WEB PROGRAMIRANJE PHP2

DOKUMENTACIJA PROJEKTA

Student:

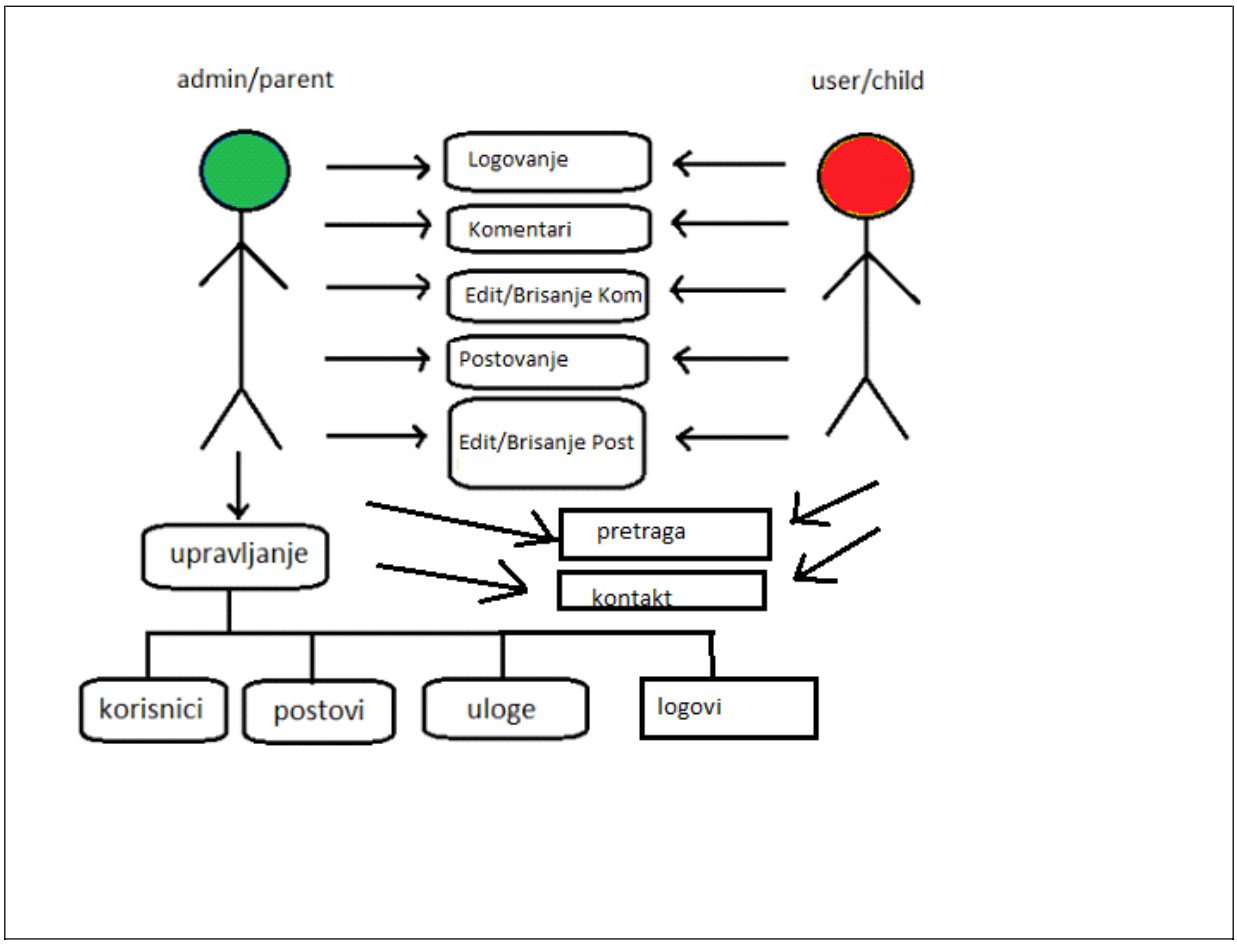
Stefan Popovic 328/15

Beograd 2021.

SADRŽAJ

1.Opis funkcionalnosti. 1
2. Skica struktura stranica 3
3. Dijagram baze podataka..... 4
3.MVC Organizacija 5
Kontroleri. 5
Modeli 6
View. 8
4. javaScript kod. 9
5. PHP kod. 10

1.Opis funkcionalnosti

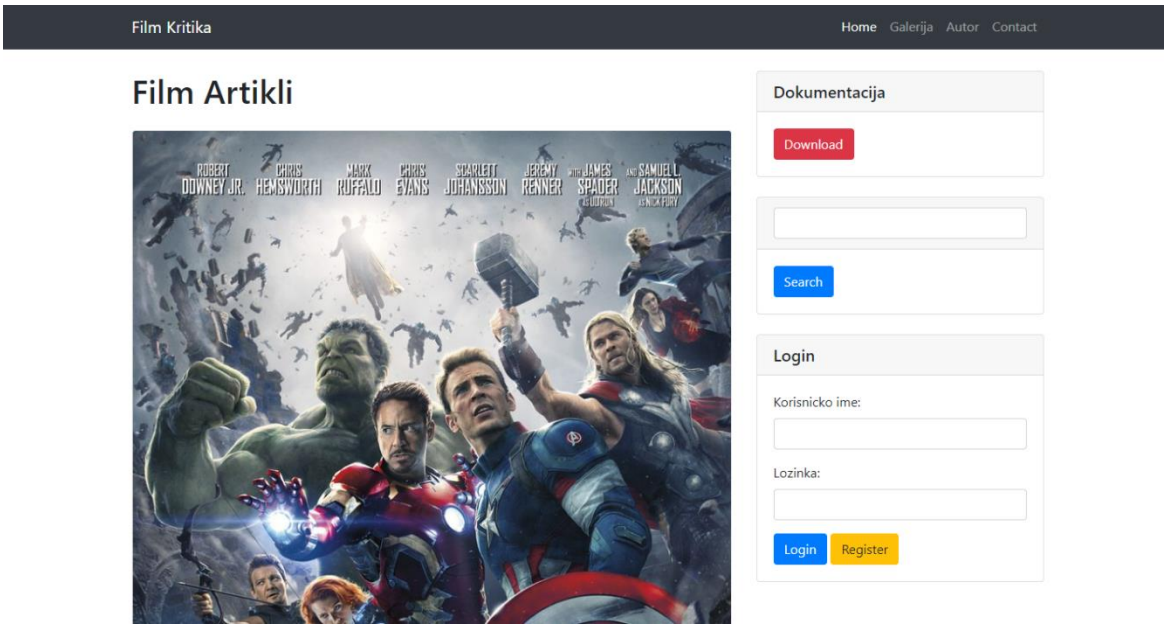


Komentare mogu da unesu samo autorizovani korisnici, pored toga mogu da obrisu i promene sopstvene komentare, dok administrator moze svaki komentar da obrise ali samo svoje da menja i da unese sopstveni komentar. Autorizovani korisnici mogu menjaju i brisu svoje postove dok administrator moze svacije. Administrator upravlja celim sadrzajem na sajtu i to: korisnicima, postovima, i ulogama I ima pristup logovima.

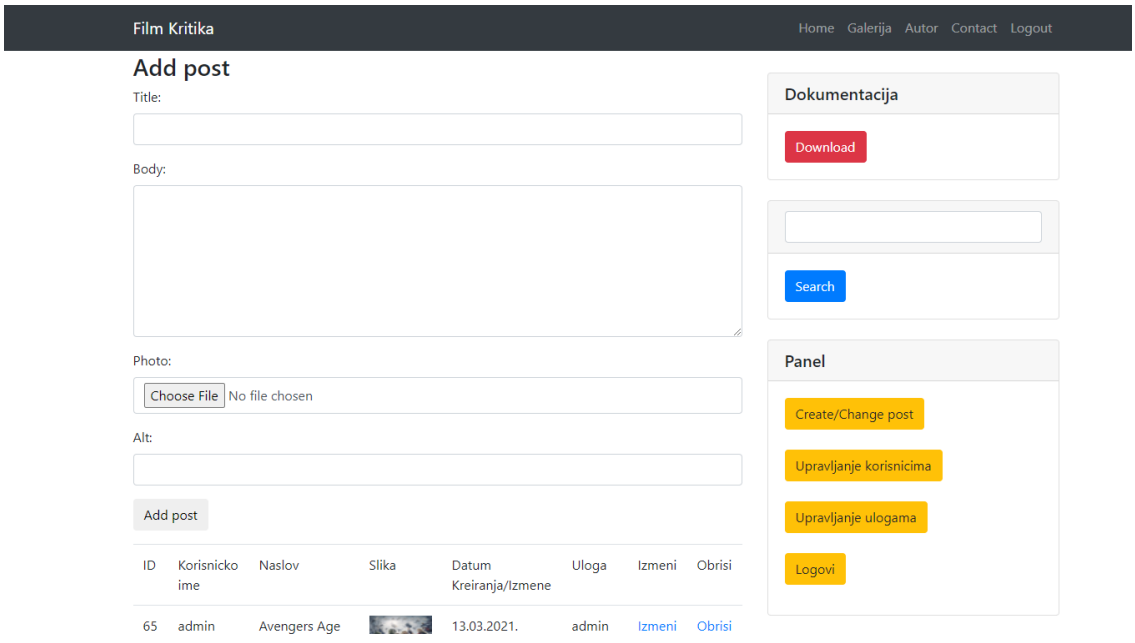
Koriščeni jezici	• HTML 4 i HTML 5 • CSS • javaScript • PHP • MySQL
Korišćeni HTML šabloni (framework-ci)	https://startbootstrap.com/template-overviews/blog-home/
Koriščene biblioteke i njihovi url-ovi odakle ste ih preuzeli	• jQuery • Bootstrap • Sve biblioteke koje su koriscene su vec bile u okviru template-a
Korišćene biblioteke iz framework-a	• Laravel
Korišćene php biblioteke van fremework-a	
Koje funkcionalnosti su realizove putem AJAX-a	Editovanje komentara
Navesti lokacije gde je napisan AJAX kod (fajlovi ili delovi html stranice)	public/js/ajax.js
Navesti lokacije gde je napisan sopstveni javaScript kod (fajlovi ili delovi html stranice)	public/js/provera.js
URL sajta/aplikacije	
Pristupni parametri	Admin/Admin1 i banana/Banana1

2. Skica struktura stranica

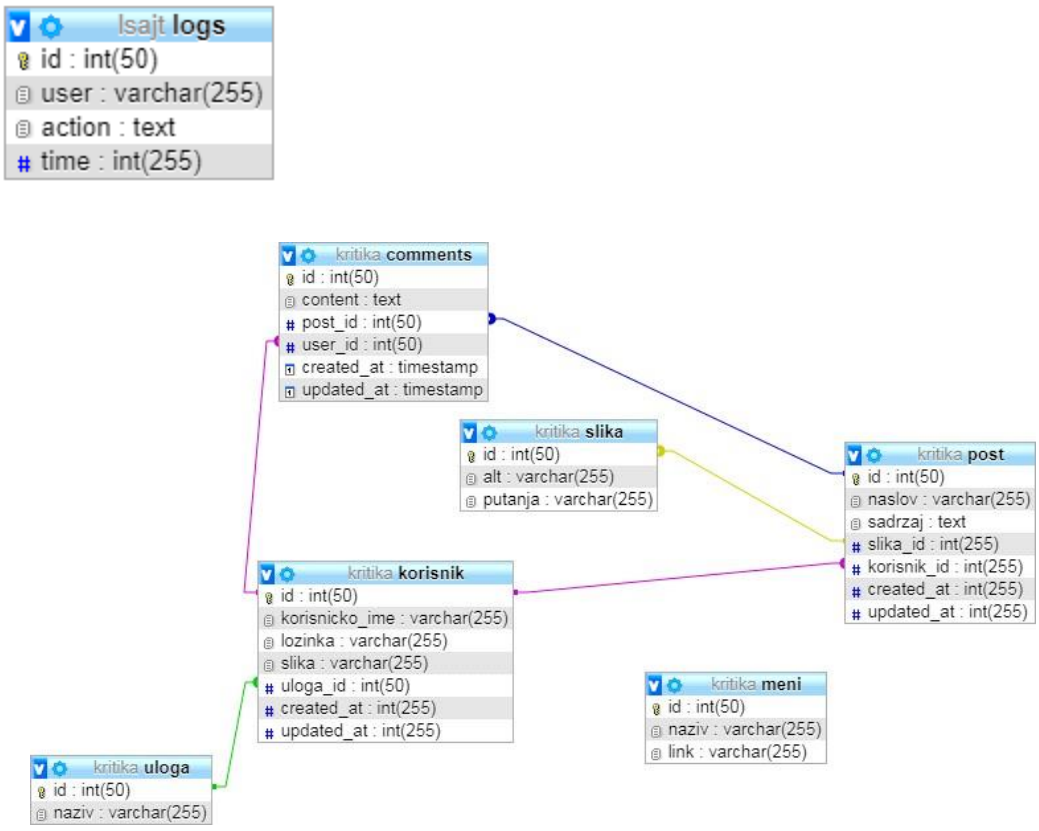
Pocetna strana:



Admin strana:



3. Dijagram baze podataka



3.MVC Organizacija

Kontroleri

CommentsController
postComment(Request $request, $postId)
editComment(Request $request, $commentId)
deleteComment($commentId)
show($commentId)

FrontendController
__construct()
index()
autor()
galerija()
getPost($id)
download()
Search()
Showlogs()

ContactUsController
index()
handleform(Request $request)

DateRangeController
__construct()
index(Request $request)

KorisnikController
__construct()
show($id = null)
regshow($id = null)
store(Request $request)
regstore(Request $request)
update($id, Request $request)
destroy($id)

LoginController
login(Request $request)
logout(Request $request)

PostController
__construct()
create()
store(Request $request)
postshow($id = null)
update($id, Request $request)
destroy($id)

UlogaController
__construct()
show($id = null)
store(Request $request)
update($id, Request $request)
destroy($id)

Modeli

CommentsModel
save()
update($id)
getPostComments($postId)
delete($id)
getUserComments($userId)
find($id)
canUserDeleteComment($commentId)

Korisnik
getAll()
get()
getByUsernameAndPassword()
save()
update()
delete()

Meni
getAll()

Logs
getlogs()
getlog()

Post
getAll()
save()
get($id)
update()
delete()
deleted()
getPictureId($postId)

Slika
save()
find($id)
delete($id)

Uloga
getAll()
get()

save()
update()
delete()

View

pages
autor.blade.php
contact.blade.php
createKorisnik.blade.php
createPost.blade.php
createUloga.blade.php
galerija.blade.php
learch.blade.php
logovi.blade.php
home.blade.php
post.blade.php
register.blade.php

layouts
front.blade.php

emails
email.blade.php

components
comments.blade.php
footer.blade.php
nav.blade.php
sidebar.blade.php

4. javaScript kod

provera.js

```
function proveraKontakt(){
	var ime = document.getElementById("imeprezime").value;
	var regIme = /^[a-z\d]{4,}$/;
	if(regIme.test(ime)){
		document.getElementById("imeprezime").style.borderColor = "green";
	}
	else {
		document.getElementById("imeprezime").style.borderColor = "red";
	}
	var password = document.getElementById("sifra").value;
	var regpassword = /^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)[a-zA-Z\d]{6,}$/;
	if(regpassword.test(password)){
		document.getElementById("sifra").style.borderColor = "green";
	}
	else {
		document.getElementById("sifra").style.borderColor = "red";
	}


}
```

Ajax.js

```
function editComment(id,pid) {
        var content=$('#kom'+id).text();
        document.getElementById("txta").value=content;
        var url=document.getElementById("pform").action;
        var res = url.replace(pid, id);
        var form = document.getElementById('pform');
        form.action=res;
        $.ajax({
            method : "get",
            processData: false, contentType: false,
            data: {
                'id' : id,
                'content' : content,
            },
            type : 'json',
            success : function(data) {
                showComment(id,content,form);
            },
            error : function(error)
            {
                console.log("Something went wrong....");
            }
        });
}

function showComment(id,content,form) {
    var url="/editComment/" + id + "/edit";
    var editForm = form;
    $("#comment"+id).html(editForm);
}

function updateComment(id) {
    var editedData = $("#commentContent").val();
    $.ajaxSetup({
        headers: {
            'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')
        }
    });
    $.ajax({
        "url" :"/comments/" + id + "/edit",
        method : "post",
        data : {
            'content' : editedData,
            '_method' : "post",
            'csrf-token' : X-CSRF-TOKEN
        },
        type : 'json',
        success : function(data) {
            hideAndReset(id, editedData);
        },
        error : function(error)
```

```
        {
            console.log("Something went wrong....");
        }
    });
}

function hideAndReset(id, editedData)
{
    console.log(editedData);
    $("#comment" + id).html(editedData);
    $("#comments").html("<div class='alert alert-info'>Comment successfully
edited!</div>");
}
```

5. PHP kod

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, $postId)
    {
        switch($request->akcija) {
            case 'post':
        if ($request->get("content")) {
            $commentsModel = new CommentsModel();
            $commentsModel->content = $request->get("content");
            $commentsModel->post_id = $postId;
            try {
                $commentsModel->save();
                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.");
        break;
    }
```

```
    }

    public function editComment(Request $request, $commentId)
    {
        switch($request->akcija) {
            case 'edit':
        $commentModel = new CommentsModel();
        try {
            $commentModel->content = $request->get('content1');
            $commentModel->update($commentId);
            return redirect()->back()->with('success', "Komentar uspesno izmenjen.");
        } catch (\Exception $e) {
            \Log::error("Greska pri izmeni komentara " . $e->getMessage());
            return redirect()->back()->with('warning', "Doslo je do greske na serveru.");
        }
        break;
    }

    }

    public function deleteComment($commentId)
    {
        $commentModel = new CommentsModel();
        if ($commentModel->find($commentId)) {
            try {
                if ($commentModel->delete($commentId)) {
                    return redirect()->back()->with("success", "Comment successfully
deleted.");
                } else {
                    \Log::critical("Korisnik bez dozvoljenih prava pokusao da obrise
komentar.");
                    return redirect()->back()->with("warning", "An error has occurred,
please try again later.");
                }
            } catch (QueryException $e) {
                \Log::error("Greska pri brisanju komentara  " . $e->getMessage());
                return redirect()->back()->with("warning", "An error has occurred, please
try again later.");
            }
        }
        return redirect()->back();
    }

    public function show($commentId)
    {
        $commentModel = new CommentsModel();
        $this->data['comments'] = $commentModel->getPostComments($commentId);

        if(!empty($commentId)){
            $commentModel->id = $commentId;
            $this->data['comment'] = $commentModel->getPostComments($commentId);
        return view('components.comments', $this->data);

    }
```

```
                }
}
```

FrontendController.php

```
<?php

namespace App\Http\Controllers;
use App\Models\CommentsModel;
use App\Models\Meni;
use App\Models\Post;
use App\Models\Logs;
use App\Models\Korisnik;
use Illuminate\Http\Request;
use DB;

class FrontendController extends Controller {

    private $data = [];

    public function __construct(){
        $meni = new Meni();
        $this->data['menus'] = $meni->getAll();
    }
    public function index(){
        $post = new Post();
        $this->data['posts'] =$post->getAll();

        $users = DB::table('post')->select('*','post.id as postId','post.created_at as
create','post.updated_at as update')->join('slika as s','post.slika_id','=','s.id')-
>join('korisnik as k','post.korisnik_id','=','k.id')->join('uloga as
u','k.uloga_id','=','u.id')->orderBy('post.created_at','desc');
        $this->data['users'] =$users->simplePaginate(5);

        return view('pages.home', $this->data);
    }
    public function autor(){
        $korisnik = new Korisnik();
        $this->data['korisnici'] = $korisnik->getAll();
        return view('pages.autor', $this->data);
    }
     public function galerija(){
        $post = new Post();
        $this->data['posts'] =$post->getAll();
        return view('pages.galerija', $this->data);
    }
    public function showlogs(){
        $logs = new Logs();
        $this->data['logovi'] = $logs->getlogs();
        return view('pages.logovi' , $this->data);
    }
```

```
    public function search(){
        $keyword = request('search');
        $post = new Post();
        $this->data['posts'] =$post->getAll();

        $users = DB::table('post')->select('*','post.id as postId','post.created_at as
create','post.updated_at as update')->join('slika as s','post.slika_id','=','s.id')-
>join('korisnik as k','post.korisnik_id','=','k.id')->join('uloga as
u','k.uloga_id','=','u.id')->orderBy('post.created_at','desc');
        $this->data['users'] =$users->where('post.naslov','like','%'.$keyword.'%')-
>simplePaginate(5);
        return view('pages.home', $this->data);
    }
    public function getPost($id){
        $post = new Post();
        $this->data['singlePost'] = $post->get($id);

        $commentModel = new CommentsModel();
        $this->data['comments'] = $commentModel->getPostComments($id);

        return view('pages.post', $this->data);
    }
    public function download(){
        $headers = array(
          'Content-Type: application/pdf',
        );
        return response()->download(('dokumentacija.pdf'), 'dokumentacija.pdf', $headers);
    }

}
```

KorisnikController.php

```
<?php

namespace App\Http\Controllers;
use App\Models\Meni;
use App\Models\Uloga;
use App\Models\Korisnik;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\File;

class KorisnikController extends Controller {

    private $data = [];

    public function __construct() {
        $uloga = new Uloga();
        $this->data['uloge'] = $uloga->getAll();
                $meni = new Meni();
                $this->data['menus'] = $meni->getAll();
```

```
    }

    public function show($id = null){
        $korisnik = new Korisnik();
        $this->data['korisnici'] = $korisnik->getAll();

        if(!empty($id)){
            $korisnik->id = $id;
            $this->data['korisnik'] = $korisnik->get();
        }

        return view('pages.createKorisnik', $this->data);
    }
        public function regshow($id = null){
        $korisnik = new Korisnik();
        $this->data['korisnici'] = $korisnik->getAll();

        if(!empty($id)){
            $korisnik->id = $id;
            $this->data['korisnik'] = $korisnik->get();
        }

        return view('pages.register', $this->data);
    }

    public function store(Request $request){

                $rules=[
            'korisnickoIme' => 'required','unique:korisnik,korisnicko_ime','min:4',
                        'ddlUloga' => 'required','not_in:0',
                        'lozinka' => [
                            'required',
                            'min:6',
                            'regex:/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)[a-zA-Z\d]{6,}$/'],
            'slika' => 'required','mimes:jpg,jpeg,png,gif'
        ];

                $messages = [
                "lozinka.regex" => 'Password must contain one uppercase letter and one
number.'];

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


        $korisnicko_ime = $request->get('korisnickoIme');
        $lozinka = $request->get('lozinka');
        $uloga_id = $request->get('ddlUloga');

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

        $tmp_putanja = $slika->getPathName();
        $ekstenzija = $slika->getClientOriginalExtension();
        $ime_fajla = time().'.'.$ekstenzija;
```

```
        $putanja = 'images/'.$ime_fajla;

        $putanja_server = ($putanja);

        try {
            File::move($tmp_putanja, $putanja_server);


            $korisnik = new Korisnik();
            $korisnik->korisnicko_ime = $korisnicko_ime;
            $korisnik->lozinka = $lozinka;
            $korisnik->slika = $putanja;
            $korisnik->uloga_id = $uloga_id;

            $rez = $korisnik->save();
            if($rez == 1){
                return redirect('/users')->with('message','Uspesan unos!');
            }
            else {
                return redirect('/users')->with('message','Greska pri unosu!');
            }
        }
        catch (\Exception $ex){
            \Log::error('MOJA GRESKA: '.$ex->getMessage());
        }
    }
        public function regstore(Request $request){

        $rules=[
            'korisnickoIme' => ['required','unique:korisnik,korisnicko_ime','min:4'],
                        'lozinka' => [
                            'required',
                            'min:6',
                            'regex:/^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)[a-zA-Z\d]{6,}$/'],
            'slika' => 'required','mimes:jpg,jpeg,png,gif'
        ];

                $messages = [
                "lozinka.regex" => 'Password must contain one uppercase letter and one
number.'];

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

        $korisnicko_ime = $request->get('korisnickoIme');
        $lozinka = $request->get('lozinka');
        $uloga_id = "2";

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

        $tmp_putanja = $slika->getPathName();
        $ekstenzija = $slika->getClientOriginalExtension();
        $ime_fajla = time().'.'.$ekstenzija;
        $putanja = 'images/'.$ime_fajla;
```

```
        $putanja_server = ($putanja);
        try {
            File::move($tmp_putanja, $putanja_server);


            $korisnik = new Korisnik();
            $korisnik->korisnicko_ime = $korisnicko_ime;
            $korisnik->lozinka = $lozinka;
            $korisnik->slika = $putanja;
            $korisnik->uloga_id = $uloga_id;

            $rez = $korisnik->save();

            if($rez == 1){
                return redirect('/register')->with('message','Uspesan unos!');
            }
            else {
                return redirect('/register')->with('message','Greska pri unosu!');
            }
        }
        catch (\Exception $ex){
            \Log::error('MOJA GRESKA: '.$ex->getMessage());
        }
    }

    public function update($id, Request $request) {
        $korisnicko_ime = $request->get('korisnickoIme');
        $lozinka = $request->get('lozinka');
        $uloga_id = $request->get('ddlUloga');

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

        $korisnik = new Korisnik();
        $korisnik->id = $id;
        $korisnik->korisnicko_ime = $korisnicko_ime;
        $korisnik->lozinka = $lozinka;
        $korisnik->uloga_id = $uloga_id;

        if(!empty($slika)){


            $korisnik_to_update = $korisnik->get();
            File::delete($korisnik_to_update->slika);


            $tmp_putanja = $slika->getPathName();
            $ime_fajla = time().'.'.$slika->getClientOriginalExtension();
            $putanja = 'images/'.$ime_fajla;
            $putanja_server = ($putanja);

            File::move($tmp_putanja, $putanja_server);

            $korisnik->slika = $putanja;
```

```
        }

        $rez = $korisnik->update();

        if($rez == 1){
            return redirect('/users')->with('message','Uspesan update!');
        }
        else {
            return redirect('/users')->with('message','Greska pri update-u!');
        }
    }

    public function destroy($id){
        $korisnik = new Korisnik();
        $korisnik->id = $id;

        $korisnik_to_update = $korisnik->get();
        File::delete($korisnik_to_update->slika);

        $rez = $korisnik->delete();
        if($rez == 1){
            return redirect('/users')->with('message','Uspesan delete!');
        }
        else {
            return redirect('/users')->with('message','Greska pri delete-u!');
        }
    }
}
```

LoginController.php

```
<?php

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

class LoginController extends Controller {

    public function login(Request $request){
        $request->validate([
            'tbKorisnickoIme' => ['required','alpha'],
        ], [
            'required' => 'Polje :attribute je obavezno!'
        ]);

        $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('/');
    }
}
```

PostController.php

```
<?php

namespace App\Http\Controllers;
use Validator;
use App\Models\CommentsModel;
use App\Models\Meni;
use App\Models\Post;
use App\Models\Slika;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\File;



class PostController extends Controller {

    private $data = [];

    public function __construct(){
        $meni = new Meni();
        $this->data['menus'] = $meni->getAll();
    }

    public function create(){

        return view('pages.createPost', $this->data);
    }
```

```php
    public function store(Request $request) {


        $rules = [
            'title' => 'regex:/^[A-Z][a-z]+(\s[\w\d\-]+)*$/',
            'body' => 'required',
            'photo' => 'required|mimes:jpg,jpeg,png,gif|max:3000',
            'alt' => 'required'
        ];
        $custom_messages = [
            'required' => 'Polje :attribute je obavezno!',
            'title.regex' => 'Polje title treba da pocne velikim slovom!',
            '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;
        $new_path = ($folder).$file_name;

        try {


            File::move($tmp_path, $new_path);



            $slika = new Slika();
            $slika->alt = trim($request->get('alt'));
            $slika->putanja = 'images/'.$file_name;
            $slika_id = $slika->save();


            $post = new Post();
            $post->naslov = $request->get('title');
            $post->sadrzaj = $request->get('body');
            $post->korisnik_id = session()->get('user')[0]->id;
            $post->slika_id = $slika_id;
            $post->save();

            return redirect('/')->with('success','Uspesno ste dodali post i sliku!');
        }
        catch(\Illuminate\Database\QueryException $ex){
            \Log::error($ex->getMessage());
            return redirect()->back()->with('error','Greska pri dodavanju posta u bazu!');
        }
        catch(\Symfony\Component\HttpFoundation\File\Exception\FileException $ex) {
            \Log::error('Problem sa fajlom!! '.$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..');
        }
    }
        public function postshow($id = null){
        $post = new Post();
        $this->data['postovi'] = $post->getAll();

        if(!empty($id)){
            $post->id = $id;
            $this->data['post'] = $post->get($id);
        }

        return view('pages.createPost', $this->data);
    }

        public function update(Request $request,$id) {
                $rules = [
            'title' => 'regex:/^[A-Z][a-z]+(\s[\w\d\-]+)*$/',
            'body' => 'required',
            'photo' => '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);


            $oldPictureId = null;

            $photo = $request->file('photo');
            $post = new post();
            $post->id = $id;
            $post->naslov = $request->get('title');
            $post->sadrzaj = $request->get('body');
            $post->slika_id=$post->get($id)->slika_id;

            $oldPictureId = $post->getPictureId($id);

            if (!empty($photo)){

                            try{

                                $rezz=$post->deleted();
                                $folder = ('images/');
                                $extension = $photo->getClientOriginalExtension();
                                $file_name = time().".".$extension;
```

```
                                $photo->move($folder, $file_name);

                                $slika = new Slika();
                                $slika->putanja = 'images/' . $file_name;
                                $slika->alt = trim($request->get('alt'));
                                $slika_id = $slika->save();
                                $post->slika_id = $slika_id;


                            }catch (QueryException $e) {
                        \Log::error("Greska pri update-u objave: " . $e->getMessage());
                    } catch (FileException $e) {
                        \Log::error("Greska pri update-u objave u dodavanju slike: " . $e-
>getMessage());
                    }
                   }
                    $rez=$post->update();

                    if($rez == 1){
                    return redirect('/posts')->with("success", "Post successfully edited!");
                }else {
                    return redirect('/posts')->with('message','Greska pri update-u!');
                }


                }




        public function destroy($id){
        $post = new Post();
        $post->id = $id;

                $post->slika_id=$post->get($id)->slika_id;

                $rezz=$post->deleted();


        $rez = $post->delete();
        if($rez == 1 && $rezz==1){
            return redirect('/posts')->with('message','Uspesan delete!');
        }
        else {
            return redirect('/posts')->with('message','Greska pri delete-u!');
        }
    }
}
```

UlogaController.php

```php
<?php

namespace App\Http\Controllers;
use App\Models\Meni;
use App\Models\Uloga;
use Illuminate\Http\Request;

class UlogaController extends Controller
{
    private $data = [];
    private $model = null;

        public function __construct(){
        $meni = new Meni();
        $this->data['menus'] = $meni->getAll();
    }

    public function show($id = null){
        $uloga = new Uloga();
        $this->data['roles'] = $uloga->getAll();

        if(!empty($id)){
            $uloga->id = $id;
            $this->data['updateRole'] = $uloga->get();
        }
        return view('pages.createUloga', $this->data);
    }

    public function store(Request $request){
        $request->validate([
            'naziv' => [
                'required', 'unique:uloga'
            ]
        ], [
            'required' => 'Polje :attribute je obavezno',
            'unique' => 'Polje :attribute mora biti jedinstveno u bazi.'
        ]);

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

        $uloga = new Uloga();
        $uloga->naziv = $naziv;

        $rez = $uloga->save();

        if($rez == 1){
            return redirect()->back()->with('success', 'Dodata uloga!');
        }
        else {
            return redirect()->back()->with('error','Greska pri dodavanju uloge!');
        }
```

```
    }

    public function update($id, Request $request){
        $naziv = $request->get('naziv');

        $uloga = new Uloga();
        $uloga->id = $id;
        $uloga->naziv = $naziv;

        $rez = $uloga->update();

        if($rez == 1){
            return redirect()->back()->with('success', 'Azurirana uloga!');
        }
        else {
            return redirect()->back()->with('error','Greska pri azuriranju uloge!');
        }

    }

    public function destroy($id){
        $uloga = new Uloga();
        $uloga->id = $id;
        $rez = $uloga->delete();

        if($rez == 1){
            return redirect()->back()->with('success', 'Obrisana uloga!');
        }
        else {
            return redirect()->back()->with('error','Greska pri brisanju uloge!');
        }
    }
}
```

DateRangeController.php

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Meni;
use App\Models\Logs;
use DB;

class DateRangeController extends Controller
{
    private $data = [];
    public function __construct(){
      $meni = new Meni();
      $this->data['menus'] = $meni->getAll();
    }
    function index(Request $request)
```

```
    {
        $logs = new Logs();
      if(!empty($request->fromdate))
      {
        $fromdate=$request->fromdate;
        $todate=$request->todate;
         $this->data['logovi']=$logs->getlog($fromdate, $todate);
      }
      else
      {
      $this->data['logovi'] = $logs->getlogs();
      }

     return view('pages.logovi',$this->data);
    }
}

?>
```

ContactUsController.php

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Http\Requests;
use App\Models\Meni;
use Mail;

class ContactUsController extends Controller
{
    public function index()
    {
        $meni = new Meni();
        $this->data['menus'] = $meni->getAll();
        return view('pages.contact',$this->data);
    }

    public function handleForm(Request $request)
    {

        $this->validate($request, ['name' => 'required', 'email' => 'required|email',
'message_body' => 'required|min:20']);

        $data = ['name' => $request->get('name') , 'email' => $request->get('email') ,
'messageBody' => $request->get('message_body') ];

        Mail::send('emails.email', $data, function ($message) use ($data)
        {
            $message->from($data['email'], $data['name']);
            $message->to('androidxx8@gmail.com', 'Admin')
                ->subject('Laravel Sajt');
        });

        return redirect()
```

```
            ->back()
            ->with('success', 'Thank you for your feedback');

    }

}
```

Middleware/CheckUserRole.php

<?php

namespace App\Http\Middleware;

use Closure;

class CheckUserRole

{

    /**

     * 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')){
            $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!');
    }
}
```

CommentsModel.php

```php
<?php

namespace App\Models;


class CommentsModel
{
    public $content;
```

```php
    public $post_id;
    private $table = 'comments';

    public function save()
    {
        return \DB::table($this->table)
            ->insert([
               'user_id' => session()->get('user')[0]->id,
               'post_id' => $this->post_id,
                'content' => $this->content,
                'created_at' => date("Y-m-d H:i:s"),
                'updated_at' => date("Y-m-d H:i:s")
            ]);
    }

    public function update($id)
    {
        return \DB::table($this->table)
            ->where('id', $id)
            ->update([
               'content' => $this->content,
               'updated_at' => date("Y-m-d H:i:s")
            ]);
    }

    public function getPostComments($postId)
    {
        return \DB::table($this->table)
            ->join("korisnik", "comments.user_id", "=", "korisnik.id")
            ->where('post_id', $postId)
            ->select('comments.*', 'korisnik.korisnicko_ime')
            ->orderBy('comments.created_at','desc')
            ->get();
    }

    public function delete($id)
    {
        if($this->canUserDeleteComment($id)) {
            return \DB::table($this->table)->delete($id);
        }
        return 0;
    }

    public function getUserComments($userId)
    {
        return \DB::table($this->table)->where('user_id', $userId)->get();
    }

    public function find($id)
    {
        return \DB::table($this->table)
            ->where('id', $id)->get()->first();
    }
    private function canUserDeleteComment($commentId)
    {
```

```
        $comment = $this->find($commentId);
        return $comment ? (session()->get('user')[0]->naziv == 'admin') || ($comment-
>user_id == session()->get('user')[0]->id) : false;
    }
}
```

Korisnik.php

<?php

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

class Korisnik
{
	public $id;
	public $korisnicko_ime;
	public $lozinka;
	public $slika;
	public $uloga_id;

	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 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),
                'slika' => $this->slika,
                'uloga_id' => $this->uloga_id,
        'created_at' => time()
        ]);
        return $rez;
}
```

```
public function update(){
        $data = [
                'korisnicko_ime' => $this->korisnicko_ime,
                'lozinka' => md5($this->lozinka),
                'uloga_id' => $this->uloga_id,
        'updated_at' => time(),
        ];

        if(!empty($this->slika)){ // ako je upload-ovana slika
                $data['slika'] = $this->slika;
        }

        $rez = DB::table('korisnik')
        ->where('id',$this->id)
                        ->update($data)
                        ;
        return $rez;
}

public function delete(){
        $rezultat = DB::table('korisnik')
                                ->where('id', $this->id)
                                ->delete();
        return $rezultat;
}
```

}

Meni.php

<?php

namespace App\Models;

use Illuminate\Support\Facades\DB;

class Meni {
    //put your code here

    public function getAll(){
        $rezultat = DB::table('meni')->get();
        return $rezultat;
    }
}

Logs.php

```php
<?php

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

class Logs {
    public $fromdate;
    public $todate;

    public function getlogs(){
        $rezultat = DB::table('logs')->get();
        return $rezultat;
    }
    public function getlog($fromdate,$todate){
        $newfromdate=$fromdate.' 00:00:00';
```

```
        $newtodate=$todate. '23:59:59';
        $newfrom=\Carbon\Carbon::parse($newfromdate)->timestamp;
        $newto=\Carbon\Carbon::parse($newtodate)->timestamp;
        $rezultat = DB::table('logs')
        ->select('*')
        ->whereBetween('time',array($newfrom,$newto))
        ->get();
        return $rezultat;
    }
}
```

Post.php

<?php

namespace App\Models;

use Illuminate\Support\Facades\DB;

```
class Post {
        public $id;
        public $naslov;
        public $sadrzaj;
        public $korisnik_id;
        public $slika_id;
      private $table = 'post';

        public function getAll(){
                $rezultat = DB::table('post')
                                        ->select('*','post.id as postId','post.created_at as
create','post.updated_at as update')
                                        ->join('slika as s','post.slika_id','=','s.id')
                                        ->join('korisnik as k','post.korisnik_id','=','k.id')
                        ->join('uloga as u','k.uloga_id','=','u.id')
                                        ->orderBy('post.created_at','desc')
                                        ->get();
                return $rezultat;
        }

        public function save() {
                $rezultat = DB::table('post')->insert([
```

```
                'naslov' => $this->naslov,
                'sadrzaj' => $this->sadrzaj,
                'korisnik_id' => $this->korisnik_id,
                'created_at' => time(),
                'slika_id' => $this->slika_id
            ]);
            return $rezultat;
        }


public function get($id){
    $rezultat =
        DB::table($this->table)
        ->select('*',
            'post.id AS postId',
            'post.created_at as create',
            'post.updated_at as update',
            'korisnik.korisnicko_ime as postKorisnik',
            'k.korisnicko_ime as komentarKorisnik',
            'comments.post_id as comments')
        ->join('slika','slika.id','=','post.slika_id')
        // Komentari se mogu dohvatati i posebnim upitom
        ->join('korisnik','korisnik.id','=','post.korisnik_id')
        ->leftJoin('comments','comments.post_id','=','post.id')
        ->leftJoin('korisnik AS k','k.id','=','comments.user_id')
```

```
        ->where('post.id','=',$id )
        ->first();
    return $rezultat;
}
public function update(){
            $data = [
                    'naslov' => $this->naslov,
                    'sadrzaj' => $this->sadrzaj,
                    'alt' => $this->alt,
            'updated_at' => time(),
            ];

        if ($this->slika_id != null) {
        $data['slika_id'] = $this->slika_id;
        }

            $rez = DB::table('post')
            ->where('id',$this->id)
                        ->update($data)
                        ;
            return $rez;

    }
```

```
        public function delete(){
                $rezultat = DB::table('post')
                                        ->where('id', $this->id)
                                        ->delete();
                return $rezultat;
        }
    public function deleted(){
                $rezultat = DB::table('slika')
                                        ->where('id', $this->slika_id)
                                        ->delete();
                return $rezultat;
        }
    public function getPictureId($postId)
  {
    return \DB::table($this->table)
        ->where('slika_id', $postId->slika_id)
        ->select("slika.id as id")
        ->get()->first()->id;
  }

}
```

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 $id;
        }
    public function find($id)
  {
    return \DB::table($this->table)
        ->where('id', $id)
        ->get()
        ->first();
  }
```

```
    public function delete($id)
    {
        return \DB::table($this->table)
            ->delete($id);
    }
}
```

Uloga.php

```
<?php

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

class Uloga {

        private $table = 'uloga';
        public $id;
        public $naziv;

        public function getAll(){
                $rezultat = DB::table($this->table)->get();
                return $rezultat;
        }
```

```
public function get(){
        return DB::table($this->table)
                        ->select('*')
                        ->where('id',$this->id)
                        ->first();
}

public function save(){
        return DB::table($this->table)
                ->insert([
                        'naziv' => $this->naziv
                ]);
}

public function update(){
        return DB::table($this->table)
                ->where('id', $this->id)
                ->update([
                        'naziv' => $this->naziv
                ]);
}

public function delete(){
        return DB::table($this->table)
```

```
                    ->where('id', $this->id)
                    ->delete()
                ;
        }


}
```

comments.blade.php

```
<div class="comments heading">
    <h3>Comments</h3>
    @foreach($comments as $comment)
        <div class="media">
            <div class="media-body">
                <table class="table">
                    <tr>
                        <td>
                <h4 class="media-heading">  {{ $comment->korisnicko_ime }}</h4>
                <h6><label>{{ date("F j, Y", strtotime($comment->created_at))
}}</label></h6>
                <span id="comment{{$comment->id}}">
                    <span id="kom{{$comment->id}}">{{ $comment->content }}</span></br>
                    @if(session('user'))
                        @if(session()->get('user')[0]->id == $comment->user_id ||
session()->get('user')[0]->naziv == 'admin')
                            <a href="{{ route("deleteComment", ['id' => $comment->id])
}}"><i class="fa fa-trash">Obrisi</i></a>
                        @endif
                        @if(session()->get('user')[0]->id == $comment->user_id)
                                <a href="#comment{{$comment->id}}"><i class="fa fa-edit"
onclick="editComment({{ $comment->id }},{{ $singlePost->postId}})">Izmeni</i></a>
                        @endif
                    @endif
                    <span id="#comments"></span>
                </span>
                        </td>
                </tr>
                </table>
            </div>
            <div class="media-right">

            </div>
        </div>
    @endforeach
```

```
</div>
```

nav.blade.php

```
<!-- Navigacija -->
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top">
      <div class="container">
        <a class="navbar-brand" href="">Film Kritika</a>
```

```
<button class="navbar-toggler" 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">
  <ul class="navbar-nav ml-auto">
    @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'))

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

</nav>

<!--// Navigacija -->

sidebar.blade.php

```
<!-- Desna strana -->
        <div class="col-md-4">
          <div class="card my-4">
            <h5 class="card-header">Dokumentacija</h5>
            <div class="card-body">
              <div class="input-group">
                   <a href="{{ asset('/download') }}" class="btn btn-danger">Download</a>
              </div>
            </div>
          </div>
          <form action="{{ route('search') }}" method="GET">
          {{ csrf_field() }}
          <div class="card my-4">
            <h5 class="card-header"><input type="text" id="search" name="search"
class="form-control"/></h5>
            <div class="card-body">
              <div class="input-group">
              <input type="submit" name="btnSearch" value="Search" class="btn btn-
primary"/>
              </div>
            </div>
          </div>
          </form>
            @if(!session()->has('user'))
          <div class="card my-4">
            <h5 class="card-header">Login</h5>
            <div class="card-body">
              <div class="row">
                <div class="col-lg-12">


                  @isset($errors)
                    @if($errors->any())
                      @foreach($errors->all() as $error)
                        {{ $error }}
                      @endforeach
                    @endif
                  @endisset

                  <!-- LOGIN FORM -->
```

```
                <form action="{{ route('login') }}" method="POST">
                  {{ csrf_field() }}
                  <div class="form-group">
                      <label>Korisnicko ime:</label>
                      <input type="text" id="imeprezime" name="tbKorisnickoIme"
class="form-control" onkeyup="proveraKontakt()"/>
                  </div>

                  <div class="form-group">
                    <label>Lozinka:</label>
                    <input type="password" id="sifra" name="tbLozinka" class="form-
control" onkeyup="proveraKontakt()"/>
                  </div>

                  <input type="submit" name="btnLogin" value="Login" class="btn btn-
primary"/>
                <a href="{{ asset('register') }}" class="btn btn-warning">Register</a>
                </form>

                <!--// LOGIN FORM -->

              </div>
            </div>
          </div>
        </div>
          @endif
        @if(session()->has('user'))
        <div class="card my-4">
          <h5 class="card-header">Panel</h5>
          <div class="card-body">
            <p>
              @if(session()->get('user')[0]->naziv == 'admin')
              <a href="{{ asset('/posts') }}" class="btn btn-warning">Create/Change
post</a>
              <br/> <br/>
              <a href="{{ asset('/users') }}" class="btn btn-warning">Upravljanje
korisnicima</a>
              <br/> <br/>
              <a href="{{ asset('/roles') }}" class="btn btn-warning">Upravljanje
ulogama</a>
              <br/> <br/>
              <a href="{{ asset('/logs') }}" class="btn btn-warning">Logovi</a>
              @endif
              @if(session()->get('user')[0]->naziv != 'admin')
              <a href="{{ asset('/posts/create') }}" class="btn btn-
warning">Create/Change post</a>
              @endif
            </p>
          </div>
        </div>
        @endif
      </div>
  <!--// Desna strana -->
```

front.blade.php

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <link rel="icon" type="image/jpg" href="images/film.png" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="kritike filmove,ocene i misljenja">
    <meta name="keywords" content="filmovi,kritika,ocene,forum" />
    <meta name="author" content="Stefan Popovic">

    <title> Film Kritike | @yield('title') </title>

    @section('appendCss')

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

<script src="{{ asset('/') }}vendor/jquery/jquery.min.js"></script>
<link rel="stylesheet" href="{{ asset('/') }}js/Popup/dist/magnific-popup.css">
<script src="{{ asset('/') }}js/provera.js"></script>
<script src="{{ asset('/') }}js/Popup/dist/jquery.magnific-popup.min.js"></script>
<script type="text/javascript">
  $(document).ready(function() {

    $('.image-popup-vertical-fit').magnificPopup({
      type: 'image',
      closeOnContentClick: true,
      mainClass: 'mfp-img-mobile',
      image: {
        verticalFit: true
      }

    });

    $('.image-popup-fit-width').magnificPopup({
      type: 'image',
      closeOnContentClick: true,
      image: {
        verticalFit: false
      }
    });

    $('.image-popup-no-margins').magnificPopup({
      type: 'image',
      closeOnContentClick: true,
      closeBtnInside: false,
      fixedContentPos: true,
    mainClass: 'mfp-no-margins mfp-with-zoom',
    image: {
      verticalFit: true
    },
```

```
    zoom: {
      enabled: true,
      duration: 300
    }
  });

  });


</script>

  </head>
  <body>

    @include('components.nav')

    <div class="container">

      <div class="row">
        <div class="col-lg-12"></div>
        <div class="col-lg-12">
          @empty(!session('error'))
            <div class="alert alert-danger">{{ session('error') }}</div>
          @endempty

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

        @yield('content')

        @include('components.sidebar')

      </div>

    </div>


    @include('components.footer')


    @section('appendJavascript')


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

  </body>

</html>
```

autor.blade.php

```
@extends('layouts.front')

@section('title')
    Autor page
@endsection

@section('appendCss')
    @parent
```

```
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection
@section('content')
<!-- Sadrzaj -->
        <div class="col-md-8">

          <h1 class="my-4">O Autoru
          </h1>

          <!-- Blog Post -->
          <div class="card mb-4">
             @isset($korisnici)
             @foreach($korisnici as $korisnik)
             @if($korisnik->korisnikId==9)
            <img class="card-img-top" src="{{ asset($korisnik->slika) }}" alt="autor" >@endif
             @endforeach
           @endisset
          </div>
          <table class="table">
               <tr>
                 <td>Ime i Prezime:</td>
                 <td>Stefan Popovic</td>
               </tr>
               <tr>
```

```
            <td>Datum rodjenja:</td>
                <td>02.04.1994</td>
            </tr>
            <tr>
                <td>Skola:</td>
                <td>Visoka ICT</td>
            </tr>
            <tr>
            <td>Email:</td>
                <td>stefan.popovic.328.15@ict.edu.rs</td>
            </tr>
            <td></td><td></td>
        </table>


    </div>


@endsection
```

createKorisnik.blade.php

```
@extends('layouts.front')
```

```
@section('title')
   Korisnik create
@endsection

@section('appendCss')
   @parent
   <!-- Custom styles for this template -->
   <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->
      <div class="col-md-8">
         <h3>Add korisnik</h3>

         @empty(!session('message'))
          {{ session('message') }}
         @endempty

         @isset($errors)
          @if($errors->any())
            @foreach($errors->all() as $error)
              <div class="alert alert-danger"> {{ $error }} </div>
            @endforeach
          @endif
```

```
@endisset

<form action="{{ (isset($korisnik))? asset('/users/update/'.$korisnik->id) : asset('/users/store') }}" method="POST" enctype="multipart/form-data">

 {{ csrf_field() }}

 <div class="form-group">
  <label>Korisnicko ime:</label>
  <input type="text" name="korisnickoIme" class="form-control" value="{{ (isset($korisnik))? $korisnik->korisnicko_ime : old('korisnickoIme') }}"/>
 </div>
 <div class="form-group">
  <label>Lozinka:</label>
  <input type="password" name="lozinka" class="form-control" value="{{ (isset($korisnik))? $korisnik->lozinka : old('lozinka') }}"/>
 </div>
 <div class="form-group">
  <label>Slika:</label>

@isset($korisnik)
<img src="{{ asset($korisnik->slika) }}" width="35" height="35" style="border-radius: 500px; -moz-border-radius: 500px; margin-top: 2px;"/>
  @endisset
```

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


          </div>
          <div class="form-group">
            <label>Uloga:</label>
            <select name="ddlUloga">
              <option value="0">Izaberite</option>


              @foreach($uloge as $uloga)
                <option value="{{ $uloga->id }}" {{ (isset($korisnik) && $korisnik->uloga_id == $uloga->id)? 'selected' : (old('ddlUloga')==$uloga->id)? 'selected' : '' }} > {{ $uloga->naziv }} </option>
              @endforeach


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


        <table class="table">
          <tr>
            <td>ID</td>
            <td>Korisnicko ime</td>
```

```
            <td>Slika</td>
            <td>Datum Kreiranja/Izmene</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> {{ $korisnik->korisnicko_ime }} </td>
              <td> <img src="{{ asset($korisnik->slika) }}" width="75" height="75"
style="border-radius: 500px; -moz-border-radius: 500px;" /> </td>
              <td> {{ date("d.m.Y. H:i:s", $korisnik->created_at) }}</br>@if($korisnik-
>updated_at!=null){{ date("d.m.Y. H:i:s", $korisnik->updated_at) }}@endif </td>
              <td> {{ $korisnik->naziv }} </td>
              <td> <a href="{{ asset('/users/'.$korisnik->korisnikId) }}">Izmeni</a> </td>
              <td> <a href="{{ asset('/users/destroy/'.$korisnik->korisnikId) }}">Obrisi</a>
</td>
            </tr>
          @endforeach
          @endisset
        </table>
      </div>
            <!--// Sadrzaj -->
```

@endsection

createPost.blade.php

```
@extends('layouts.front')

@section('title')
    Post create
@endsection

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->
        <div class="col-md-8">
            <h3>Add post</h3>

            @isset($errors)
              @if($errors->any())
                @foreach($errors->all() as $error)
                  <div class="alert alert-danger"> {{ $error }} </div>
                @endforeach
              @endif
            @endisset
            <form action="{{ (isset($post))? asset('/posts/update/'.$post->postId) :
asset('/posts/store') }}" method="POST" enctype="multipart/form-data">
              {{ csrf_field() }}
               {{ method_field('PUT')}}
              <div class="form-group">
                <label>Title:</label>
                <input type="text" name="title" class="form-control" value="{{
(isset($post))? $post->naslov : old('naslov') }}"/>
              </div>
              <div class="form-group">
                <label>Body:</label>
                <textarea name="body" class="form-control" rows="7">{{ (isset($post))?
$post->sadrzaj : old('sadrzaj') }}</textarea>
              </div>
              <div class="form-group">
                <label>Photo:</label>
                @isset($post)
                <img src="{{ asset($post->putanja) }}" width="35" height="35"/>
                @endisset
                <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="{{
(isset($post))? $post->alt : old('alt') }}"/>
              </div>
              <div class="form-group">
                <input type="submit" name="addPost" value="{{ (isset($post))? 'Change post'
: 'Add post' }}" class="btn btn-default" />
              </div>
            </form>

            <table class="table">
                <tr>
                  <td>ID</td>
                  <td>Korisnicko ime</td>
                  <td>Naslov</td>
                  <td>Slika</td>
                  <td>Datum Kreiranja/Izmene</td>
                  <td>Uloga</td>
                  <td>Izmeni</td>
                  <td>Obrisi</td>
                </tr>
                <!-- Prikaz postova-->
                @isset($postovi)
                @if(session()->get('user')[0]->naziv == 'admin')
                @foreach($postovi as $post)
                  <tr>
                    <td> {{ $post->postId }} </td>
                    <td> {{ $post->korisnicko_ime }} </td>
                    <td> {{ $post->naslov }} </td>
                    <td> <img src="{{ asset($post->putanja) }}" width="75" height="75" />
</td>
                    <td> {{ date("d.m.Y. H:i:s", $post->create) }}</br>@if($post-
>update!=null){{ date("d.m.Y. H:i:s", $post->update) }}@endif </td>
                    <td> {{ $post->naziv }} </td>
                    <td> <a href="{{ asset('/posts/'.$post->postId) }}">Izmeni</a> </td>
                    <td> <a href="{{ asset('/posts/destroy/'.$post->postId) }}">Obrisi</a>
</td>
                  </tr>
                @endforeach
                @endif
                @if(session()->get('user')[0]->naziv != 'admin')
                @foreach($postovi->where("korisnik_id", "=", session()->get('user')[0]->id)
as $post)
                  <tr>
                    <td> {{ $post->postId }} </td>
                    <td> {{ $post->korisnicko_ime }} </td>
                    <td> {{ $post->naslov }} </td>
                    <td> <img src="{{ asset($post->putanja) }}" width="75" height="75" />
</td>
                    <td> {{ date("d.m.Y. H:i:s", $post->create) }}</br>@if($post-
>update!=null){{ date("d.m.Y. H:i:s", $post->update) }}@endif </td>
                    <td> {{ $post->naziv }} </td>
                    <td> <a href="{{ asset('/posts/'.$post->postId) }}">Izmeni</a> </td>
```

```
                    <td> <a href="{{ asset('/posts/destroy/'.$post->postId) }}">Obrisi</a>
</td>
                  </tr>
                @endforeach
                @endif
                @endisset
            </table>

        </div>
    <!--// Sadrzaj -->
@endsection
```

createUloga.blade.php

@extends('layouts.front')

@section('title')

Manage users

@endsection

@section('appendCss')

@parent

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

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

@endsection

@section('content')

<!-- Sadrzaj -->

<h3>Manage users</h3>

```
@isset($errors)
  @if($errors->any())
    @foreach($errors->all() as $error)
      <div class="alert alert-danger"> {{ $error }} </div>
    @endforeach
  @endif
@endisset

<form action="{{ isset($updateRole)? asset('/roles/update/'.$updateRole->id) : asset('/roles/store') }}" method="POST">
  {{ csrf_field() }}
  <div class="form-group">
    <label>Naziv:</label>
    <input type="text" name="naziv" class="form-control" value="{{ isset($updateRole)? $updateRole->naziv : old('naziv') }}"/>
  </div>
  <div class="form-group">
    <input type="submit" name="addUloga" value="{{ isset($updateRole)? 'Change role' : 'Add uloga' }}" class="btn btn-default" />
  </div>
</form>

<table class="table">
  <tr>
    <th>ID</th>
```

```
            <th>Naziv</th>
            <th>Izmeni</th>
            <th>Obrisi</th>
          </tr>

          @foreach($roles as $role)
            <tr>
              <td>{{ $role->id }}</td>
              <td>{{ $role->naziv }}</td>
              <td><a href="{{ asset('/roles/'.$role->id) }}">Izmeni</a></td>
              <td><a href="{{ asset('/roles/destroy/'.$role->id) }}">Obrisi</a></td>
            </tr>
          @endforeach
        </table>
      </div>
                <!--// Sadrzaj -->
@endsection
```

galerija.blade.php

```
@extends('layouts.front')

@section('title')
   Galerija
@endsection
```

```
@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->

            <div class="col-md-8">
        <div class="gallery-top heading"><br>
            <h3>Nasa galerija</h3><br>
        </div>
            <div class="card mb-4"></div>
        <div class="container">
        <div class="row">
        @isset($posts)
        @foreach($posts as $post)
        <div class="col-sm-3" style="height: 200px">
        <a class="image-popup-no-margins" href="{{ asset('/').$post->putanja }}"><img src="{{ asset('/').$post->putanja }}" class="img img-thumbnail" alt="{{ $post->alt }}"></a>
        <br>
        <center><strong>{{ $post->alt }}</strong></center>
            </div>
```

```
            @endforeach
            @endisset
            </div>
            </div>
                <div class="card mb-4"></div>
        </div>
@endsection
```

home.blade.php

```
@extends('layouts.front')

@section('title')
    Home page
@endsection

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->
        <div class="col-md-8">
```

```
<h1 class="my-4">Film Artikli


</h1>


@isset($users)
{{-- var_dump($posts) --}}


@foreach($users as $post)
  {{-- var_dump($post) --}}
<!-- Blog Post -->
<div class="card mb-4">
  <img class="card-img-top" src="{{ asset('/').$post->putanja }}" alt="{{ $post->alt }}">
  <div class="card-body">
    <h2 class="card-title">{{ $post->naslov }}</h2>
    <p class="card-text">
       {{ $post->sadrzaj }}
    </p>
    <a href="{{ asset('/post/'.$post->postId )}}" class="btn btn-primary">Read More
&rarr;</a>
  </div>
  <div class="card-footer text-muted">
    Posted on {{ date("d.m.Y. H:i:s", $post->create) }} by
    <a href="#">{{ $post->korisnicko_ime }}</a>
  </div>
```

```
        </div>
                <!--// Blog Post -->
            @endforeach
            @endisset

        <!-- Pagination -->
        <ul class="pagination justify-content-center mb-4">
          <li class="page-item">
            {{ $users->links() }}
          </li>
        </ul>
      </div>
                <!--// Sadrzaj -->
@endsection
```

post.blade.php

```
@extends('layouts.front')

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection
@section('content')
```

```
@isset($singlePost)
<!-- Sadrzaj -->
    <div class="col-md-8">


     <!-- Title -->
     <h1 class="mt-4">{{ $singlePost->naslov }}</h1>

     <!-- Author -->
     <p class="lead">
       by
       <a href="#">{{ $singlePost->postKorisnik }}</a>
     </p>

     <hr>

     <!-- Date/Time -->
     <p>Posted on {{ date("d.m.Y. H:i:s", $singlePost->create) }}</p>

     <hr>

     <!-- Preview Image -->
     <img class="img-fluid rounded" src="{{ asset('/'.$singlePost->putanja)}}" alt="{{
$singlePost->alt}}">
```

```
<hr>

<!-- Post Content -->
<p> {{ $singlePost->sadrzaj }}</p>

            <hr>

<!-- Comments Form -->
@if(session()->get('user'))
<div class="card my-4">
  <h5 class="card-header">Leave a Comment:</h5>
  <div class="card-body">
    <form action="{{ route("postComment", ['postId' => $singlePost->postId]) }}" method="post">
      {{ csrf_field() }}
      <div class="form-group">
        <textarea class="form-control" rows="3" name="content"></textarea>
      </div>
      <button type="submit" class="btn btn-primary">Submit</button>
    </form>
      <br>
      @if(session('success'))
        <div class="alert alert-success">
        {{ session('success') }}
```

```
            </div>
        @endif
        @if(session('warning'))
            <div class="alert alert-warning">
            {{ session('warning') }}
            </div>
        @endif
    </div>
</div>
@else
<h4>Da bi komentarisali morate da se ulogujete.</h4>
@endif
                <!--// Comments Form -->

<!-- Single Comment -->

    @isset($comments)
    @include('components.comments')
    @endisset




                <!--// Single Comment -->

</div>
```

<!--// Sadrzaj -->

@endisset

@endsection

email.blade.php

```
Message from : {{ $name }}
<p> Name: {{ $name }}</p>
<p>E-mail: {{ $email }}</p>
<p>Message: {{ $messageBody }}</p>
```

logovi.blade.php

```
@extends('layouts.front')

@section('title')
    Logovi
@endsection

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->
        <div class="col-md-8">
            <h3>Log Activity Lists</h3>

            @empty(!session('message'))
              {{ session('message') }}
            @endempty

            @isset($errors)
              @if($errors->any())
                @foreach($errors->all() as $error)
                  <div class="alert alert-danger"> {{ $error }} </div>
                @endforeach
              @endif
            @endisset

             <br />
            <div class="row input-daterange">
                <form action="{{ asset('/flogs') }}" method="GET">
                    <label>Date from:</label>
                    <input type="date" id="from_date" name="fromdate">
                    <label>Date to:</label>
                    <input type="date" id="to_date" name="todate">
                    <input type="submit" class="btn btn-primary" value="Filter">
                </form>
            </div>
```

```
            <br />

            <table class="table table-bordered">
        <tr>
            <th>Id</th>
            <th>User</th>
            <th>Action</th>
            <th>Date</th>
        </tr>
            @isset($logovi)
            @foreach($logovi as $log)
            <tr>
                <td>{{ $log->id }}</td>
                <td>{{ $log->user }}</td>
                <td class="text-success">{{ $log->action }}</td>
                <td class="text-success">{{ date("d.m.Y. H:i:s", $log->time) }}</td>
            </tr>
            @endforeach
            @endisset

    </table>
        </div>
        <!--// Sadrzaj -->
@endsection
```

contact.blade.php

```
@extends('layouts.front')

@section('title')
    Contact
@endsection

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->
        <div class="col-md-8">
            <h3>Contact Us</h3>

            @empty(!session('message'))
              {{ session('message') }}
            @endempty

            @isset($errors)
              @if($errors->any())
                @foreach($errors->all() as $error)
                  <div class="alert alert-danger"> {{ $error }} </div>
                @endforeach
              @endif
            @endisset
```

```
            <form action="{{ url('contact-us')}}" method="POST">
              {{ csrf_field() }}
              <div class="form-group">
              <label>Name</label>
                <input type="text" name="name" class="form-control">
              </div>
              <div class="form-group">
              <label>E-mail</label>
                <input type="email" name="email" class="form-control">
              </div>
              <div class="form-group">
              <label>Message</label>
                <textarea name="message_body" class="form-control"></textarea>
              </div>
              <div class="form-group">
              <button type="submit" class="btn btn-primary">Submit</button>
              </div>
            </form>
        </div>
    <!--// Sadrzaj -->
@endsection
```

search.blade.php

```
@extends('layouts.front')

@section('title')
    Home page
@endsection

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
<!-- Sadrzaj -->
        <div class="col-md-8">

          <h1 class="my-4">Film Artikli

          </h1>

          @isset($users)
          {{-- var_dump($posts) --}}

          @foreach($users as $post)
            {{-- var_dump($post) --}}
          <!-- Blog Post -->
          <div class="card mb-4">
            <img class="card-img-top" src="{{ asset('/').$post->putanja }}" alt="{{ $post-
>alt }}">
            <div class="card-body">
              <h2 class="card-title">{{ $post->naslov }}</h2>
```

```
              <p class="card-text">
                  {{ $post->sadrzaj }}
              </p>
              <a href="{{ asset('/post/'.$post->postId )}}" class="btn btn-primary">Read
More &rarr;</a>
            </div>
            <div class="card-footer text-muted">
              Posted on {{ date("d.m.Y. H:i:s", $post->create) }} by
              <a href="#">{{ $post->korisnicko_ime }}</a>
            </div>
          </div>
    <!--// Blog Post -->
                @endforeach
                @endisset

          <!-- Pagination -->
          <ul class="pagination justify-content-center mb-4">
            <li class="page-item">
              {{ $users->appends(['search' => request('search')])->links() }}
            </li>
          </ul>
        </div>
    <!--// Sadrzaj -->
@endsection
```

register.blade.php

```
@extends('layouts.front')

@section('title')
    Korisnik create
@endsection

@section('appendCss')
    @parent
    <!-- Custom styles for this template -->
    <link href="{{ asset('/') }}css/blog-home.css" rel="stylesheet"/>
@endsection

@section('content')
```

```
<!-- Sadrzaj -->
        <div class="col-md-8">
            <h3>Add korisnik</h3>


            @empty(!session('message'))
              {{ session('message') }}
```

```
@endempty


@isset($errors)
  @if($errors->any())
    @foreach($errors->all() as $error)
      <div class="alert alert-danger"> {{ $error }} </div>
    @endforeach
  @endif
@endisset


<form action="{{ asset('/register/store') }}" method="POST" enctype="multipart/form-data">


  {{ csrf_field() }}


  <div class="form-group">
    <label>Korisnicko ime:</label>
    <input type="text" name="korisnickoIme" class="form-control" value="{{ (isset($korisnik))? $korisnik->korisnicko_ime : old('korisnickoIme') }}"/>
  </div>
  <div class="form-group">
    <label>Lozinka:</label>
    <input type="password" name="lozinka" class="form-control" value="{{ (isset($korisnik))? $korisnik->lozinka : old('lozinka') }}"/>
  </div>
  <div class="form-group">
```

```
            <label>Slika:</label>



        @isset($korisnik)

            <img src="{{ asset($korisnik->slika) }}" width="35" height="35" style="border-radius: 500px; -moz-border-radius: 500px; margin-top: 2px;"/>

            @endisset



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


          </div>

          <div class="form-group">

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

          </div>

        </form>

      </div>

                <!--// Sadrzaj -->

@endsection
```

routes.php

```php
<?php

Route::get('/',[FrontendController::class, 'index']);
Route::get('/autor',[FrontendController::class, 'autor']);
Route::get('/galerija',[FrontendController::class, 'galerija']);
Route::get('/search', [FrontendController::class, 'search'])->name('search');

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

```
    // Upravljanje korisnicima

    Route::get('/users/{id?}', [KorisnikController::class, 'show']);
    Route::post('/users/store', [KorisnikController::class, 'store']);
    Route::get('/logs', [FrontendController::class, 'showlogs']);
    Route::get('/flogs', [DateRangeController::class, 'index']);
    Route::post('/users/update/{id}',[KorisnikController::class, 'update']);
    Route::get('/users/destroy/{id}',[KorisnikController::class, 'destroy']);

    // Upravljanje ulogama

    Route::get('/roles/{id?}', [UlogaController::class, 'show']);
    Route::post('/roles/store', [UlogaController::class, 'store']);
    Route::post('/roles/update/{id}',[UlogaController::class, 'update']);
    Route::get('/roles/destroy/{id}',[UlogaController::class, 'destroy']);

});

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

    Route::get('/posts/{id?}', [PostController::class, 'postshow']);
    Route::put('/posts/store', [PostController::class, 'store']);
    Route::put('/posts/update/{id}',[PostController::class, 'update']);
    Route::get('/posts/destroy/{id}',[PostController::class, 'destroy']);

    /*
    Rute za upravljanje komentarima
*/

    Route::post("/comments/{postId}", [CommentsController::class, 'postComment'])-
>name("postComment");
    Route::post("/post/{postId}/comments/{commentId?}", [CommentsController::class,
'editComment'])->name("editComment");
    Route::get("/comments/{commentId}/delete", [CommentsController::class,
'deleteComment'])->name("deleteComment");
    Route::get('/logout', [LoginController::class, 'logout'])->name('logout');

});


Route::get('contact-us',[ContactUsController::class, 'index']);
Route::post('contact-us',[ContactUsController::class, 'handleForm']);




// Logovanje

Route::post('/login', [LoginController::class, 'login'])->name('login');
Route::get('/register', [KorisnikController::class, 'regshow']);
Route::post('/register/store', [KorisnikController::class, 'regstore']);


Route::get('/post/{id}',[FrontendController::class, 'getPost']);
```

```
Route::get('/download',[FrontendController::class, 'download']);
```