



WEB PROGRAMIRANJE PHP2

DOKUMENTACIJA PROJEKTA

Student:

Uroš Ćirić

Beograd 2018.

SADRŽAJ

| | |
|-----------------------------------|----|
| 1. Opis funkcionalnosti | 1 |
| 2. Skica struktura stranica | 1 |
| 3. Dijagram baze podataka | 2 |
| 3. MVC Organizacija | 3 |
| Kontroleri | 3 |
| Modeli | 4 |
| View | 6 |
| 4. JavaScript kod..... | 6 |
| Main.js..... | 6 |
| Admin_panel.js | 10 |
| 5. PHP kod..... | 15 |
| Kontroleri | 15 |
| Accounts..... | 15 |
| Admin_panel | 17 |
| ArticleController | 21 |
| Front | 22 |
| Survey..... | 24 |
| Middleware..... | 25 |
| Web.php | 25 |
| Modeli | 27 |
| Article..... | 27 |
| Cards..... | 27 |
| Memberships | 28 |
| MenuModel | 29 |
| Showcase | 29 |
| SurveyModel | 30 |
| User | 31 |
| UserManager | 33 |
| Views..... | 33 |
| Layouts/shell | 33 |
| Home | 35 |
| Register..... | 35 |

| | |
|----------------------------|----|
| Autor..... | 37 |
| Article..... | 37 |
| Admin_panel | 37 |
| components/admin_menu..... | 37 |
| components/articles | 38 |
| components/autor | 38 |
| components/cards | 40 |
| components/showcase | 41 |
| components/survey..... | 41 |
| admin/articles | 42 |
| admin/manage_users | 43 |
| admin/memberships | 44 |
| admin/showcase | 45 |
| admin/sruveyresults..... | 46 |
| admin/surveys..... | 46 |

1.Opis funkcionalnosti

Logovanje i registracija. Regularni izrazi na klijentskoj i na serverskoj strani

Dinamicni meni iz baze

Ako korisnik nije ulogovan, nudi se opcija za login ili register.
U meniju prikazan username ako je korisnik ulogovan i opcija da se izloguje.
Admin panel dostupan u meniju samo ako je korisnik moderator ili visi rank.

Galerija

Lightbox plugin.
Slike se dohvataju iz baze.
Slike se mogu dodavati i brisati u admin panelu.

Stranice

Koristi se AJAX.

Obrada izuzetaka.

Za svaku funkciju, error logovi se pisu u podrazumevani laravel.log fajl.

Anketa

Koristi se AJAX.
Prikazana je samo ako je korisnik ulogovan i ako nije vec glasao.
Anketa se moze kreirati i brisati u admin panelu preko dinamicne forme, mogu postojati vise pitanja u jednoj anketi i vise ponudjenih odgovora. Moze se odabrati radiobutton ili checkbox.
Rezultat ankete je dostupan administratoru u admin panelu.

Admin panel

Uredjivanje korisnika, clanarina, anketa, clankova i slika.
U nekim slucajevima se koristi AJAX.

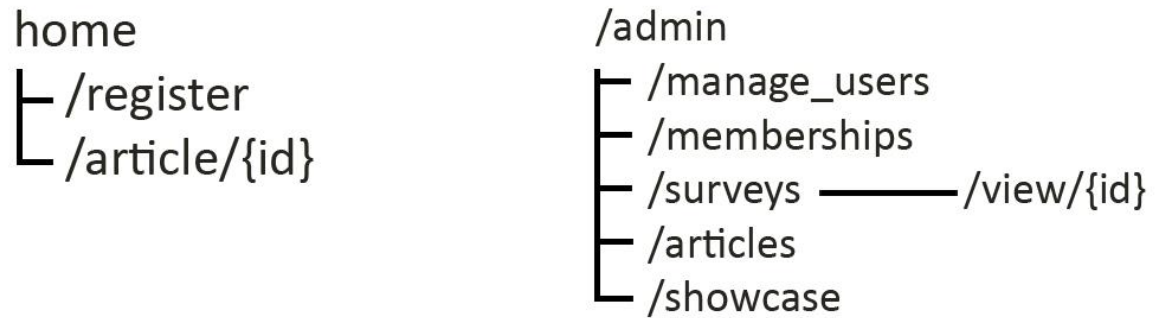
Prikaz clanarina

Clanarine se dohvataju iz baze. Mogu se uredjivati u admin panelu.

| | |
|--|---|
| Korišćeni jezici | <ul style="list-style-type: none"> • HTML, HTML5 • CSS, CSS3 • JavaScript • PHP |
| Korišćeni HTML šabloni (framework-ci) | <ul style="list-style-type: none"> • Bootstrap |
| Korišćene biblioteke i njihovi url-ovi odakle ste ih preuzeli | <ul style="list-style-type: none"> • jQuery • lightbox http://lokeshdhakar.com/projects/lightbox2/ • lightslider http://sachinchoolur.github.io/lightslider/ • Sass |
| Korišćene biblioteke iz framework-a | <ul style="list-style-type: none"> • Laravel |
| Korišćene php biblioteke van framework-a | |
| Koje funkcionalnosti su realizove putem AJAX-a | <ul style="list-style-type: none"> • Paginacija • Anketa(glasanje i kreiranje kao admin) • Editovanje i brisanje stavki iz liste kao admin |
| Navešti lokacije gde je napisan AJAX kod (fajlovi ili delovi html stranice) | |
| Navešti lokacije gde je napisan sopstveni javaScript kod (fajlovi ili delovi html stranice) | |
| URL sajta/aplikacije | http://simona-andrejic.com/ |
| Pristupni parametri | Parametri poslati na email Danijela.nikitin@ict.edu.rs |

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 možete koristiti alatku *Pencil* ili bilo koji drugi alat za kreiranje *wireframe*-a. Takođe moguće je odraditi i *print screen* strukture stranice.



3. Dijagram baze podataka

| gym nav |
|----------------------|
| id : int(11) |
| name : varchar(244) |
| route : varchar(244) |

| gym users |
|-------------------------|
| id : int(11) |
| firstname : varchar(25) |
| lastname : varchar(50) |
| username : varchar(25) |
| password : varchar(255) |
| date : date |
| email : varchar(255) |
| # phone : int(255) |
| # role_id : tinyint(1) |
| # trainer : tinyint(1) |

| gym roles |
|--------------------|
| id : int(11) |
| role : varchar(20) |

| gym cards |
|------------------------|
| id : int(11) |
| name : varchar(255) |
| color : varchar(20) |
| # price : decimal(5,2) |
| perks : text |

| gym showcase |
|----------------------|
| id : int(11) |
| title : varchar(255) |
| img : varchar(255) |
| alt : varchar(255) |

| gym articles |
|-----------------------|
| id : int(11) |
| header : varchar(255) |
| img : varchar(255) |
| alt : varchar(255) |
| text : text |

| gym surveys |
|---------------------|
| id : int(11) |
| name : varchar(255) |

| gym questions |
|-----------------------------|
| id : int(11) |
| name : varchar(255) |
| answer_format : varchar(15) |
| # survey_id : int(11) |

| gym answers |
|-------------------------|
| id : int(11) |
| # question_id : int(11) |
| name : varchar(255) |

| gym user_answers |
|-------------------------|
| id : int(11) |
| # user_id : int(11) |
| # survey_id : int(11) |
| # question_id : int(11) |
| # answer_id : int(11) |

3.MVC Organizacija

Kontroleri

| Accounts |
|-----------------------------|
| register(Request \$request) |
| login(Request \$request) |
| logout(Request \$request) |

| ArticleController |
|-------------------|
| showArticle(\$id) |

| Survey |
|---------------------------------|
| submit(Request \$request) |
| survey_delete(\$id) |
| survey_results(\$id) |
| createSurvey(Request \$request) |

| Front |
|------------------------------------|
| index(Request \$request) |
| getArticlePages(Request \$request) |
| getArticle(Request \$request) |
| register() |
| autor() |

| |
|-------------------------------------|
| Admin_panel |
| index() |
| manage_users_index() |
| memberships_index() |
| surveys_index() |
| deleteUser(Request \$request) |
| editUser(Request \$request) |
| editMembership(Request \$request) |
| deleteMembership(Request \$request) |
| insertMembership(Request \$request) |
| articles_index() |
| submitArticle(Request \$request) |
| deleteArticle(Request \$request) |
| showcase_index() |
| showcase_delete(Request \$request) |
| showcase_insert(Request \$request) |

Modeli

| |
|--|
| Article |
| getAll() |
| get(\$skip, \$take) |
| getOne(\$id) |
| insertArticle(\$header,\$img,\$alt,\$text) |
| deleteArticle(\$id) |

| |
|-----------|
| Cards |
| get_all() |

| |
|--|
| Memberships |
| get() |
| editMembership(\$id, \$name, \$color, \$price) |
| deleteMembership(\$id) |
| insertMemberstip(\$name,\$color,\$price,\$perks) |

| |
|-------------|
| MenuModel |
| getHome() |
| getAnchor() |

| |
|------------------------------------|
| Showcase |
| getAll() |
| deleteShowcase(\$id) |
| insetShowcase(\$title,\$img,\$alt) |

| |
|--|
| SurveyModel |
| getSurvey(\$user_id) |
| deleteSurvey(\$id) |
| get(\$table) |
| voteCheck(\$user_id, \$survey_id) |
| submitAnswer(\$user_id, \$survey_id, \$question_id, \$answer_id) |
| getQuestions(\$id) |
| getSurveyName(\$id) |
| insertSurvey(\$surveyName, \$surveyQuestions, \$surveyAnswers) |

| |
|-------------------------|
| User |
| register() |
| loginUser() |
| deleteUser(\$uid) |
| editUser(\$uid, \$role) |

| |
|-------------|
| UserManager |
| get() |
| get_roles() |

View

| |
|----------------------------------|
| Spisak view fajlova (.blade.php) |
| welcome |
| Home |
| Rehister |
| Autor |
| Article |
| Admin_panel |
| /layouts/shell |
| /components/admin_menu |
| /components/articles |
| /components/autor |
| /components/cards |
| /components/nav |
| /components/showcase |
| /components/survey |
| /admin/articles |
| /admin/manage_users |
| /admin/memberships |
| /admin/showcase |
| /admin/surveyresults |
| /admin/surveys |

4. javaScript kod

Main.js

```
$(document).ready(function () {  
    //Slow anchor scroll  
    $(document).on('click', 'a[href^="#"]', function (event) {  
        event.preventDefault();  
  
        $('html, body').animate({  
            scrollTop: $($.attr(this, 'href')).offset().top  
        }, 800);  
    });  
    //!Slow anchor scroll  
  
    //Form check  
    function validate(input) {  
        input.removeClass('input_warning');  
        $('.form_warning').remove();  
        $('.registerholder > button').prop('disabled', false);  
        var val = input.val();  
        var name = input.attr('name');  
        var reg;  
        var msg;
```



```

});
//!Form check

//Login dropdown
$('body').on('click', '.login_toggle', function () {
    $('#login_form').toggle(250, 'swing');
});
//!Login dropdown

//AJAX survey submit
$('body').on('click', '.survey_submit', function () {

    var survey_id = $(this).data('survey-id');

    var token = $('form[data-survey-id="' + survey_id + '>input[name="_token"]').val();
    var survey = [];

    $("input[data-survey-id=" + survey_id + "']").each(function (i, input) {
        var question_id = $(input).data('question-id');
        if ($(input).prop('checked')) {

            var answer_id = parseInt($(this).val());
            var question = [question_id, answer_id];
            survey.push(question);

        }
    });

    survey = JSON.stringify(survey);

    $.ajax({
        method: "POST",
        url: thisUrl + "/survey",
        data: {
            survey_id: survey_id,
            user_id: userID,
            survey: survey,
            _token: token,
        },
        success: function () {
            location.reload();
        }
    });
});

//!AJAX survey submit

//AJAX pager
var pagesTotal;
if ($('#pagerToken').length != 0) {
    $.ajax({

```

```

        method: "POST",
        url: baseUrl + "/articlePage",
        data: {
            _token: $('#pagerToken').val(),
        },
        success: function (data) {
            pagesTotal = data / 2;
        }
    });
}
var page = 1;
$('.articleBtnLeft').click(function () {
    page--;
    $('#articlePageNumber').html('page:' + page);
});
$('.articleBtnRight').click(function () {
    page++;
    $('#articlePageNumber').html('page:' + page);
});
$('.articleBtn').click(function () {
    if (page < 1) {
        page = 1;
        $('#articlePageNumber').html('page:' + page);
    };
    if (page > pagesTotal) {
        page = pagesTotal;
        $('#articlePageNumber').html('page:' + page);
    }
    $.ajax({
        method: "POST",
        url: baseUrl + "/getArticle",
        data: {
            page: page - 1,
            _token: $('#pagerToken').val(),
        },
        success: function (data) {
            var articles = "";
            $.each(data, function (i) {
                articles += `<article>
                    <h3>` + data[i].header + `</h3>
                    <img src="" + baseUrl + `/img/articles/` + data[i].img + ` alt="" +
data[i].alt + `">
                    <p>` + data[i].text + `</p>
                    <a class="btn btn-primary" href="" + baseUrl + `/article/` + data[i].id
+ `">Read more</a>
                </article>`;
            });
            $('#articles').html(articles);
        }
    });
});

```

```

    })
    //!AJAX pager
});

```

Admin_panel.js

```

window.onload = function () {

    //Survey create

    $('#makeSurvey').click(function () {
        var surveyName = $('#surveyName').val();
        var token = $('#surveyForm>input[name="_token"]').val();

        var surveyQuestions = [];
        var surveyAnswers = [];

        $('.questionRow').each(function (index) {
            var question = [];
            var indeks = index + 1;
            question[0] = $(this).find('option:selected').text();
            question[1] = $(this).find('.surveyQuestion').val();
            surveyQuestions.push(question);
        });
        surveyQuestions = JSON.stringify(surveyQuestions);

        $('.surveyAnswer').each(function () {
            var answer = [];
            answer[0] = $(this).data().questionnumber - 1;
            answer[1] = $(this).val();
            surveyAnswers.push(answer);
        });
        surveyAnswers = JSON.stringify(surveyAnswers);

        $.ajax({
            method: "POST",
            url: thisUrl + "/createSurvey",
            data: {
                surveyName: surveyName,
                surveyQuestions: surveyQuestions,
                surveyAnswers: surveyAnswers,
                _token: token,
            },
            success: function () {

            }
        });
    });

    //Survey controls

```

```

$('body').on('click', '.answerRow>.answerDelete', function () {
    $(this).parent().remove();
});
$('body').on('click', '.answerRow>.answerAdd', function () {
    var questionNumber = $(this).parent().data().questionnumber;
    var newAnswer = `

11


```



```

//Delete user
$('#btnDeleteUser').click(function () {
    var uid = $(this).data().uid;
    var token = $('token[data-uid="' + uid + '"]').val();
    $.ajax({
        method: "POST",
        url: baseUrl + "/admin/deleteuser",
        data: {
            uid: uid,
            _token: token,
        },
        success: function () {
            location.reload();
        }
    });
});

//!Delete user

//!Edit user
$('#btnEditUser').click(function () {
    var uid = $(this).data().uid;
    var token = $('token[data-uid="' + uid + '"]').val();
    var role = $('select[data-uid="' + uid + '"]').val();
    $.ajax({
        method: "POST",
        url: baseUrl + "/admin/edituser",
        data: {
            uid: uid,
            role: role,
            _token: token,
        },
        success: function () {
            location.reload();
        }
    });
});

//!Edit user

//Delete membership
$('#btnDeleteMembership').click(function () {
    var mid = $(this).data().mid;
    var token = $('token[data-mid="' + mid + '"]').val();
    $.ajax({
        method: "POST",
        url: baseUrl + "/admin/deletemembership",
        data: {
            id: mid,
            _token: token,
        },
    });
});

```

```

        success: function () {
            location.reload();
        }
    });

});
//!Delete membership

//Edit membership
$('.btnEditMembership').click(function () {
    var mid = $(this).data().mid;
    var token = $('#token[data-mid="' + mid + '"]').val();
    var membershipName = $('#membershipName[data-mid="' + mid + '"]').val();
    var membershipColor = $('#membershipColor[data-mid="' + mid + '"]').val();
    var membershipPrice = $('#membershipPrice[data-mid="' + mid + '"]').val();
    $.ajax({
        method: "POST",
        url: baseUrl + "/admin/editmembership",
        data: {
            id: mid,
            name: membershipName,
            color: membershipColor,
            price: membershipPrice,
            _token: token,
        },
        success: function () {
            location.reload();
        }
    });

});
//!Edit membership

//Delete showcase
$('.deleteShowcase').click(function () {
    var id = $(this).data().id;
    var token = $('#token').val();
    $.ajax({
        method: "POST",
        url: baseUrl + "/admin/deleteShowcase",
        data: {
            id: id,
            _token: token,
        },
        success: function () {
            location.reload();
        }
    });

});
//!Delete showcase

```

```
//Delete article
$('.deleteArticle').click(function () {
    var id = $(this).data().id;
    var token = $('#token').val();
    $.ajax({
        method: "POST",
        url: baseUrl + "/admin/deleteArticle",
        data: {
            id: id,
            _token: token,
        },
        success: function () {
            location.reload();
        }
    });
});

});
//!Delete article
};
```



```

        'regex' => 'Invalid input :attribute',
        'numeric' => ':attribute must be a number.',
        'email' => 'Bad email format.'
    ];
    $request->validate($rules, $custom_messages);

    //Unos u bazu
    $user = new User();
    $user->firstname = $request->get('tbFirstName');
    $user->lastname = $request->get('tbLastName');
    $user->username = $request->get('tbUsername');
    $user->password = $request->get('tbPassword');
    $user->date = $request->get('dateYear').'-'. $request->get('dateMonth').'-'. $request-
>get('dateDay');
    $user->email = $request->get('tbEmail');
    $user->phone = $request->get('tbPhone');
    try{
        $user->register();
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
    $this->data['alerts']=$user->alerts;
    if(empty($this->data['alerts'])){
        return redirect('/')->with('success','Thank you for registering, you may log in. ');
    }
    else{
        return view('register', $this->data);
    }
}

public function login(Request $request){
    $request->validate([
        'tbUsername' => ['required'],
        'tbPassword' => ['required']
    ], [
        'required' => 'You must enter both username and password'
    ]);
    $user = new User();
    $user->username = $request->get('tbUsername');
    $user->password = $request->get('tbPassword');

    try{
        $loginUser = $user->loginUser();
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

```

```

if(!empty($loginUser)){
    try{
        $request->session()->push('user', $loginUser);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }

    return redirect()->back()->with('success','Thank you for logging in');
}
return redirect()->back()->with('error','Wrong username or password');
}

public function logout(Request $request){
    try{
        $request->session()->forget('user');
        $request->session()->flush();
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
    return redirect('/');
}
}

```

Admin_panel

<?php

```
namespace App\Http\Controllers;
```

```

use App\Models\Article;
use App\Models\Memberships;
use App\Models\Showcase;
use App\Models\SurveyModel;
use App\Models\UserManager;
use App\Models\User;
use Illuminate\Http\Request;
use App\Models\MenuModel;

```

```

class Admin_panel extends Controller
{
    public function index(){
        return view('admin_panel', $this->data);
    }

    //Manage users
    public function manage_users_index(){
        $users = new UserManager();
    }
}

```

```

$this->data['users'] = $users->get();
$this->data['roles'] = $users->get_roles();
return view ('admin.manage_users', $this->data);
}

//Memberships
public function memberships_index(){
    $memberships = new Memberships();
    $this->data['memberships'] = $memberships->get();
    return view ('admin.memberships',$this->data);
}

//Surveys
public function surveys_index(){
    $surveys = new SurveyModel();
    $this->data['surveys'] = $surveys->get('surveys');
    return view ('admin.surveys', $this->data);
}

//Delete user
public function deleteUser(Request $request){
    $uid = $request->get('uid');
    $user = new User();

    try{
        $user->deleteUser($uid);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

//Edit user
public function editUser(Request $request){
    $uid = $request->get('uid');
    $role = $request->get('role');
    $user = new User();

    try{
        $user->editUser($uid, $role);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

//Delete membership
public function editMembership(Request $request){
    $id = $request->get('id');

```

```

$name = $request->get('name');
$color = $request->get('color');
$price = $request->get('price');

$membership = new Memberships();

try{
    $membership->editMembership($id, $name, $color, $price);
}
catch(\Exception $ex){
    \Log::error('error: ' . $ex->getMessage());
    return \Redirect::back()->withErrors('Something went wrong...');
}
}

//Edit membership
public function deleteMembership(Request $request){
    $id = $request->get('id');
    $membership = new Memberships();

    try{
        $membership->deleteMembership($id);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

//Insert membership
public function insertMembership(Request $request){
    $name = $request->get('name');
    $color = $request->get('color');
    $price = $request->get('price');
    $perks = $request->get('perks');

    $perk = (explode("\\",$perks));
    $perks="";
    foreach($perk as $i=>$val){
        $perks .= '<tr><td>'.$perk[$i].'\</td>';
    }

    $membership = new Memberships();

    try{
        $membership->insertMemberstip($name,$color,$price,$perks);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

```



```

        return redirect()->back()->with('Membership created');
    }

//Articles
public function articles_index(){
    $articles = new Article();
    $this->data['articles'] = $articles->getAll();
    return view('admin.articles', $this->data);
}

public function submitArticle(Request $request){
    $header = $request->get('header');
    $img="";
    $alt = $request->get('alt');
    $text = $request->get('text');

    if ($request->hasFile('image')) {
        $image = $request->file('image');
        $name = time().'.'.$image->getClientOriginalExtension();
        $destinationPath = public_path('/img/articles');
        $image->move($destinationPath, $name);
        $img=$name;
    }

    $article = new Article();
    try{
        $article->insertArticle($header,$img,$alt,$text);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
    return redirect()->back()->with('Article created');
}

public function deleteArticle(Request $request){
    $id=$request->get('id');
    $article = new Article();
    try{
        $article->deleteArticle($id);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

//Showcase
public function showcase_index(){
    $showcase = new Showcase();

```

```

        $this->data['showcase'] = $showcase->getAll();
        return view('admin.showcase', $this->data);
    }
    public function showcase_delete(Request $request){
        $id = $request->get('id');
        $showcase = new Showcase();

        try{
            $showcase->deleteShowcase($id);
        }
        catch(\Exception $ex){
            \Log::error('error: ' . $ex->getMessage());
            return \Redirect::back()->withErrors('Something went wrong...');
        }
    }
    public function showcase_insert(Request $request){
        $title = $request->get('title');
        $img="";
        $alt = $request->get('alt');
        if ($request->hasFile('image')) {
            $image = $request->file('image');
            $name = time().'.'.$image->getClientOriginalExtension();
            $destinationPath = public_path('/img/showcase');
            $image->move($destinationPath, $name);
            $img=$name;
        }

        $showcase = new Showcase();

        try{
            $showcase->insetShowcase($title,$img,$alt);
        }
        catch(\Exception $ex){
            \Log::error('error: ' . $ex->getMessage());
            return \Redirect::back()->withErrors('Something went wrong...');
        }

        return redirect()->back()->with('Showcase created');
    }
}

```

ArticleController

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 14.3.2018
 * Time: 15:44
 */

namespace App\Http\Controllers;

```

```

use App\Models\Article;

class ArticleController extends Controller
{
    public $data = [];
    public function showArticle($id){
        $article = new Article();
        $this->data['article'] = $article->getOne($id);
        return view ('article', $this->data);
    }
}

```

Front

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 25.2.2018
 * Time: 1:10
 */

namespace App\Http\Controllers;
use App\Models\Article;
use App\Models\Cards;
use App\Models\Showcase;
use App\Models\SurveyModel;
use Illuminate\Http\Request;

class Front extends Controller
{
    private $user = false;
    public function index(Request $request)
    {
        if($request->session()->has('user')){
            $this->user=true;
        }

        //Menu
        /*$menu = new MenuModel();
        $this->data['home'] = $menu->getHome();
        $this->data['nav'] = $menu->getAnchor();*/

        //Cards
        $cards = new Cards;
        $cards = $cards->get_all();
        $this->data['cards'] = $cards;
    }
}

```

```

//Surveys
if($this->user){
    $user_id= $request->session()->get('user')[0]->id;
    $surveys = new SurveyModel();

    try{
        $this->data['questions'] = $surveys->get('questions');
        $this->data['answers'] = $surveys->get('answers');
        $this->data['surveys'] = $surveys->getSurvey($user_id);
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

//Articles
$article = new Article();
try{
    $this->data['articles'] = $article->get(0,2);
}
catch(\Exception $ex){
    \Log::error('error: ' . $ex->getMessage());
    return \Redirect::back()->withErrors('Something went wrong...');
}

//Showcase
$showcase = new Showcase();
try{
    $this->data['showcase'] = $showcase->getAll();
}
catch(\Exception $ex){
    \Log::error('error: ' . $ex->getMessage());
    return \Redirect::back()->withErrors('Something went wrong...');
}

return view('home', $this->data);
}
public function getArticlePages(Request $request){
    return \DB::table('articles')->count();
}
public function getArticle(Request $request){
    $skip = $request->get('page')*2;
    $article = new Article();
    return $article->get($skip,2);
}

public function register()
{
    return view('register', $this->data);
}

```

```

    }

    public function autor(){
        return view('autor', $this->data);
    }
}

```

Survey

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 8.3.2018
 * Time: 1:44
 */

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

class Survey extends Controller
{
    public $data = [];
    public function submit(Request $request){
        $user_id = $request->get('user_id');
        $survey_id = $request->get('survey_id');
        $survey = $request->get('survey');

        $hasvoted = new SurveyModel();
        $hasvoted = $hasvoted->voteCheck($user_id,$survey_id);

        if(!$hasvoted){
            foreach(json_decode($survey) as $s){
                $question_id = $s[0];
                $answer_id = $s[1];

                $result = new SurveyModel();
                try{
                    $result->submitAnswer($user_id, $survey_id, $question_id, $answer_id);
                }
                catch(\Exception $ex){
                    \Log::error('error: ' . $ex->getMessage());
                    return \Redirect::back()->withErrors('Something went wrong...');
                }
            }
        }
    }
    public function survey_delete($id){
        $survey = new SurveyModel;
        try{
            $survey->deleteSurvey($id);
        }
    }
}

```

```

        return redirect()->back()->with('success','Survey deleted');
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}

public function survey_results($id){
    $survey = new SurveyModel();
    $survey_name = $survey->getSurveyName($id);
    $survey_questions = $survey->getQuestions($id);

    $this->data['survey_name'] = $survey_name;
    $this->data['survey_questions'] = $survey_questions;
    $this->data['survey_answers'] = $survey->get('answers');
    $this->data['survey_results'] = $survey->get('user_answers');
    return view('admin/surveyresults', $this->data);
}

public function createSurvey(Request $request){
    $surveyName = $request->get('surveyName');
    $surveyQuestions = json_decode($request->get('surveyQuestions'));
    $surveyAnswers = json_decode($request->get('surveyAnswers'));
    $survey = new SurveyModel();
    try{
        $result = $survey->insertSurvey($surveyName, $surveyQuestions, $surveyAnswers);
        return $result;
    }
    catch(\Exception $ex){
        \Log::error('error: ' . $ex->getMessage());
        return \Redirect::back()->withErrors('Something went wrong...');
    }
}
}

```

Middleware

```

'admin' => \App\Http\Middleware\RoleCheck::class,
'logged_in' => \App\Http\Middleware\LoggedIn::class

```

Web.php

```
<?php
```

```
/*
```

```
|-----
| Web Routes
|-----
|
```

```
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
```

```

|
*/
Route::pattern('id', '[0-9]+');

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

Route::get('/register', 'Front@register')->name('register');
Route::post('/register', 'Accounts@register')->name('register_send');

Route::post('/login', 'Accounts@login')->name('login');
Route::get('/logout', 'Accounts@logout')->name('logout');

Route::get('/article/{id}', ['uses' => 'ArticleController@showArticle']);

Route::post('/articlePage', 'Front@getArticlePages');
Route::post('/getArticle', 'Front@getArticle');

Route::get('/autor', 'Front@autor')->name('autor');
//Logged in
Route::group(['middleware' => 'logged_in'], function() {
    Route::post('/survey', 'Survey@submit')->name('survey.submit');
});

//Admin panel
Route::group(['middleware' => 'admin'], function() {
    Route::get('/admin', 'Admin_panel@index')->name('admin');

    Route::get('/admin/manage_users', 'Admin_panel@manage_users_index')->name('admin.manage_users');
    Route::post('/admin/deleteuser', 'Admin_panel@deleteUser');
    Route::post('/admin/edituser', 'Admin_panel@editUser');

    Route::get('/admin/memberships', 'Admin_panel@memberships_index')->name('admin.memberships');
    Route::post('/admin/deletemembership', 'Admin_panel@deleteMembership');
    Route::post('/admin/editmembership', 'Admin_panel@editMembership');
    Route::post('/admin/insertmembership', 'Admin_panel@insertMembership')->name('insertMembership');
    Route::get('/admin/surveys', 'Admin_panel@surveys_index')->name('admin.surveys');

    Route::post('/admin/surveys/createSurvey', 'Survey@createSurvey')->name('create.survey');

    Route::get('/admin/surveys/view/{id}', ['uses' => 'Survey@survey_results' ]);
    Route::get('/admin/surveys/delete/{id}', ['uses' => 'Survey@survey_delete' ]);

    Route::get('/admin/articles', 'Admin_panel@articles_index')->name('admin.articles');
    Route::post('/admin/submitArticle', 'Admin_panel@submitArticle')->name('submitArticle');
    Route::post('/admin/deleteArticle', 'Admin_panel@deleteArticle');

```

```

Route::get('/admin/showcase', 'Admin_panel@showcase_index')-
>name('admin.showcase');
Route::post('/admin/deleteShowcase', 'Admin_panel@showcase_delete');
Route::post('/admin/insertShowcase', 'Admin_panel@showcase_insert');
});

```

Modeli

Article

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 14.3.2018
 * Time: 15:38
 */

namespace App\Models;

class Article
{
    public function getAll(){
        return \DB::table('articles')->get();
    }
    public function get($skip, $take){
        return \DB::table('articles')->skip($skip)->take($take)->get();
    }
    public function getOne($id){
        return \DB::table('articles')->where('id','=', $id)->get();
    }
    public function insertArticle($header,$img,$alt,$text){
        return \DB::table('articles')->insert([
            'header' => $header,
            'img' => $img,
            'alt' => $alt,
            'text' => $text
        ]);
    }
    public function deleteArticle($id){
        return \DB::table('articles')->where('id',$id)->delete();
    }
}

```

Cards

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 1.3.2018
 * Time: 22:53

```



```

*/

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

class Cards
{
    public function get_all(){
        return DB::table('cards')->get();
    }
}

```

Memberships

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 5.3.2018
 * Time: 23:55
 */

namespace App\Models;

class Memberships
{
    public function get(){
        return \DB::table('cards')->get();
    }

    public function editMembership($id, $name, $color, $price){
        return \DB::table('cards')
            ->where('id','=',$id)
            ->update([
                'name' => $name,
                'color' => $color,
                'price' => $price
            ]);
    }

    public function deleteMembership($id){
        return \DB::table('cards')
            ->where('id','=',$id)
            ->delete();
    }

    public function insertMemberstip($name,$color,$price,$perks){
        return \DB::table('cards')->insert([
            'name'=>$name,
            'color'=>$color,
            'price'=>$price,
            'perks'=>$perks

```

```

    });
}
}

```

MenuModel

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 14.3.2018
 * Time: 22:27
 */

```

```

namespace App\Models;

```

```

class MenuModel
{
    public function getHome(){
        return \DB::table('nav')->where('id','<', '4')->get();
    }
    public function getAnchor(){
        return \DB::table('nav')->where('id','>', '3')->get();
    }
}

```

Showcase

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 14.3.2018
 * Time: 20:48
 */

```

```

namespace App\Models;

```

```

class Showcase
{
    public function getAll(){
        return \DB::table('showcase')->get();
    }
    public function deleteShowcase($id){
        return \DB::table('showcase')->where('id', '=', $id)->delete();
    }
    public function insetShowcase($title,$img,$alt){
        return \DB::table('showcase')->insert([
            'title' => $title,
            'img' => $img,
            'alt' => $alt
        ]);
    }
}

```

```

    });
}
}

```

SurveyModel

```

<?php
/**
 * Created by PhpStorm.
 * User: Korisnik
 * Date: 9.3.2018
 * Time: 0:15
 */

namespace App\Models;

class SurveyModel
{
    public function getSurvey($user_id){
        \DB::enableQueryLog();

        $survey_id=\DB::table('user_answers')->select('survey_id')->where('user_id',$user_id)->distinct()->get();
        $ids = [];
        foreach($survey_id as $i=>$id){
            $ids[$i] = $id->survey_id;
        }
        $surveys = \DB::table('surveys')
            ->whereNotIn('id', $ids)
            ->get();

        //dd(\DB::getQueryLog());
        return $surveys;
    }

    public function deleteSurvey($id){
        \DB::table('surveys')
            ->join('questions', 'surveys.id', '=', 'questions.survey_id')
            ->join('answers', 'questions.id', '=', 'answers.question_id')
            ->where('surveys.id', '=', $id)
            ->delete();
        \DB::table('user_answers')
            ->where('survey_id', '=', $id)
            ->delete();
    }

    public function get($table){
        return \DB::table($table)->get();
    }

    public function voteCheck($user_id, $survey_id){

```

```

        return \DB::table('user_answers')
            ->where([
                ['user_id', $user_id],
                ['survey_id', $survey_id]
            ])
            ->exists();
    }

    public function submitAnswer($user_id, $survey_id, $question_id, $answer_id){
        return \DB::table('user_answers')->insert([
            'user_id' => $user_id,
            'survey_id' => $survey_id,
            'question_id' => $question_id,
            'answer_id' => $answer_id
        ]);
    }

    public function getSurveyName($id){
        return \DB::table('surveys')->select('name')->where('id', '=', $id)->get();
    }
    public function getQuestions($id){
        return \DB::table('questions')->where('survey_id', '=', $id)->get();
    }

    public function insertSurvey($surveyName, $surveyQuestions, $surveyAnswers){
        $surveyID = \DB::table('surveys')->insertGetId(['name' => $surveyName]);
        foreach($surveyQuestions as $i=>$question){
            $questionID = \DB::table('questions')->insertGetId([
                'answer_format' => $surveyQuestions[$i][0],
                'name' => $surveyQuestions[$i][1],
                'survey_id' => $surveyID
            ]);
            foreach($surveyAnswers as $j=>$answer){
                if($surveyAnswers[$j][0]==$i){
                    \DB::table('answers')->insert([
                        'name'=> $surveyAnswers[$j][1],
                        'question_id'=> $questionID
                    ]);
                }
            }
        }
    }
}

```

User

```
<?php
```

```
/**
```

```
* Created by PhpStorm.
```

```
* User: Korisnik
```

```
* Date: 26.2.2018
```

```
* Time: 22:59
```

```
*/
```

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

```
class User
```

```
{
```

```
    public $firstname;  
    public $lastname;  
    public $username;  
    public $password;  
    public $date;  
    public $email;  
    public $phone;
```

```
    public $alerts = [];
```

```
    public function register(){
```

```
        $check_username = DB::table('users')->where('username', $this->username)->count();  
        $check_email = DB::table('users')->where('email', $this->email)->count();
```

```
        if($check_username>0){
```

```
            $this->alerts[]='User with that username already exists.';
```

```
        }
```

```
        elseif($check_email>0){
```

```
            $this->alerts[]='User with that email address already exists.';
```

```
        }
```

```
        else{
```

```
            return DB::table('users')->insert([  
                'firstname' => $this->firstname,  
                'lastname' => $this->lastname,  
                'username' => $this->username,  
                'password' => md5($this->password),  
                'date' => $this->date,  
                'email' => $this->email,  
                'phone' => $this->phone
```

```
            ]);
```

```
        }
```

```
    }
```

```
    public function loginUser(){
```

```
        return DB::table('users')  
            ->select('users.*','roles.role')  
            ->join('roles','users.role_id', '=', 'roles.id')  
            ->where([  
                'username' => $this->username,  
                'password' => md5($this->password)  
            ])  
            ->first();
```

```
    }
```

```
    public function deleteUser($uid){
```

```
        return DB::table('users')->where('id','=',$uid)->delete();
```

```

    }
    public function editUser($uid, $role){
        return \DB::table('users')
            ->where('id','=',$uid)
            ->update(['role_id'=>$role]);
    }
}

```

UserManager

```
<?php
```

```
/**
```

```
 * Created by PhpStorm.
```

```
 * User: Korisnik
```

```
 * Date: 6.3.2018
```

```
 * Time: 11:49
```

```
 */
```

```
namespace App\Models;
```

```
class UserManager
```

```
{
```

```
    public function get(){
```

```
        return \DB::table('users')
```

```
            ->join('roles', 'users.role_id', '=', 'roles.id')
```

```
            ->select('users.*', 'roles.role')
```

```
            ->orderBy('id')
```

```
            ->get();
```

```
    }
```

```
    public function get_roles(){
```

```
        return \DB::table('roles')->get();
```

```
    }
```

```
}
```

Views

Layouts/shell

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<!--Language in which the website is written in (en for english, sr for serbian). For more info
visit https://en.wikipedia.org/wiki/List\_of\_ISO\_639-1\_codes-->
```

```
<head>
```

```
    <meta charset="utf-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
```

```
    <title></title>
```

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

```
    <!--Describe your website. This will be displayed as description under your website name
when searched on google.-->
```

```
    <meta name="keywords" content="">
```

```
    <!--Put keywords for your website. This will make your website more discoverable.-->
```

```

<link rel="canonical" href="" />
<!--This page's url.-->
<meta name="author" content="https://plus.google.com/111116356739335627675">
<!--This is my google plus page. Not necessary.-->
<!--Facebook meta tags-->
<meta property="fb:app_id" content="" />
<!--Facebook page id.-->
<meta property="og:url" content="">
<!--This page's url.-->
<meta property="og:image" content="">
<!--Image url. 1200 x 630 minimum-->
<meta property="og:description" content="">
<!--Description-->
<meta property="og:title" content="">
<!--Title of this page.-->
<meta property="og:site_name" content="">
<!--If this isn't the site's main page. Put the name of the main page here.-->
<!--Twitter meta tags-->
<meta name="twitter:card" content="">
<!--Explains what the card is. You will mostly use an image so put
"summary_large_image"-->
<meta name="twitter:url" content="">
<!--This page's url.-->
<meta name="twitter:title" content="">
<!--Title.-->
<meta name="twitter:description" content="">
<!--Description.-->
<meta name="twitter:image" content="">
<!--Put the same image as for facebook.-->
<!--Google plus meta tags-->
<meta itemprop="name" content="">
<!--Title.-->
<meta itemprop="description" content="">
<!--Description.-->
<meta itemprop="image" content="">
<!--Put the same image as for facebook.-->
<!--Css-->
<link rel="stylesheet" href="{ {url('/')} }/css/main.css">
@yield('css')
</head>

<body>
  @include('components.nav')
  <div class="container-fluid">
    @yield('content')
  </div>
  <!--Scripts-->
  <script src="{ {url('/')} }/js/jquery-3.3.1.min.js"></script>
  <script src="{ {url('/')} }/js/popper.min.js"></script>
  <script src="{ {url('/')} }/js/bootstrap.min.js"></script>
  <script src="{ {url('/')} }/js/main.js"></script>

```

```

    @yield('js')
<script>
    const thisUrl = "{{ Request::url() }}";
    const baseUrl = "{{ url('/') }}";
    @if(session()->has('user'))
    const userID = "{{ session()->get('user')[0]->id }}";
    @endif
</script>
</body>

</html>

```

Home

```

@extends('layouts.shell')
@section('css')
    <link rel="stylesheet" href="{{ url('/') }}/css/lightslider.min.css">
    <link rel="stylesheet" href="{{ url('/') }}/css/lightbox.min.css">
@endsection
@section('content')
    <div id="page_home" class="page">
        @include('components.cards')
    </div>
    @if(session()->has('user'))
        @if(!$surveys->isEmpty())
            <div id="page_survey" class="page">
                @include('components.survey')
            </div>
        @endif
    @endif
    <div id="page_articles" class="page">
        @include('components.articles')
    </div>
    <div id="page_showcase" class="page">
        @include('components.showcase')
    </div>
@endsection
@section('js')
    <script src="{{ url('/') }}/js/lightslider.min.js"></script>
    <script src="{{ url('/') }}/js/lightbox.min.js"></script>
@endsection

```

Register

```

@extends('layouts.shell') @section('content')
<div id="register_page">
</div>
<form action="{{ route('register_send') }}" method="POST" enctype="multipart/form-data"
class="container rounded" id="register_form">
    {{ csrf_field() }}
    <table>
        <tr>

```



```

        <td><input type="text" class="form-control" name="tbFirstName" id="tbFirstName"
placeholder="Firstname"></td>
        <td><input type="text" class="form-control" name="tbLastName" id="tbLastName"
placeholder="Lastname"></td>
    </tr>
    <tr>
        <td><input type="text" class="form-control" name="tbUsername" id="tbUsername"
placeholder="Username"></td>
        <td><input type="password" class="form-control" name="tbPassword"
id="tbPassword" placeholder="Password"></td>
    </tr>
    <tr>
        <td colspan="2">
            <p class="text-muted">Birth date (optional):</p>
            <div class="dateholder">
                <select class="form-control dateselector" id="dateDay" name="dateDay">
                    <option value="">Day</option>
                    @for($i=1; $i<=31; $i++)
                        <option value="{{ $i }}">{{ $i }}</option>
                    @endfor
                </select>
                <select class="form-control dateselector" id="dateMonth" name="dateMonth">
                    <option value="">Month</option>
                    @for($i=1; $i<=12; $i++)
                        <option value="{{ $i }}">{{ $i }}</option>
                    @endfor
                </select>
                <select class="form-control dateselector" id="dateYear" name="dateYear">
                    <option value="">Year</option>
                    @for($i=date('Y'); $i>=date('Y')-100; $i--)
                        <option value="{{ $i }}">{{ $i }}</option>
                    @endfor
                </select>
            </div>
        </td>
    </tr>
    <tr>
        <td colspan="2"><input type="email" class="form-control" name="tbEmail"
id="tbEmail" placeholder="Email"></td>
    </tr>
    <tr>
        <td colspan="2"><input type="tel" class="form-control" name="tbPhone"
id="tbPhone" placeholder="Phone number (optional)"></td>
    </tr>
    <tr class="registration_row">
        <td colspan="2" class="registerholder"><button type="submit" class="btn btn-
primary">Register</button></td>
    </tr>
</table>
</form>
@if ($errors->any() OR !empty($alerts))

```

```

<div class="container form_alerts">
  <div class="alert alert-danger">
    <ul>
      @if(!empty($alerts)) @foreach ($alerts as $alert)
        <li>{{ $alert }}</li>
      @endforeach @endif() @if($errors->any()) @foreach ($errors->all() as $error)
        <li>{{ $error }}</li>
      @endforeach @endif()
    </ul>
  </div>
</div>
@endif @endsection

```

Autor

```

@extends('layouts.shell')
@section('content')
  <div id="page_autor">
    @include('components.autor')
  </div>
@endsection

```

Article

```

@extends('layouts.shell')
@section('content')
  <div id="single_article" class="page">
    <article>
      <h3>{{ $article[0]->header }}</h3>
      alt } }}">
      <p>{{ $article[0]->text }}</p>
    </article>
  </div>
@endsection

```

Admin_panel

```

@extends('layouts.shell')
@section('css')
  <link rel="stylesheet" href="{{ {url('/')} } }/css/admin_panel.css">
@show
@section('content')
  @include('components.admin_menu')
  <div id="admin_right">
    @yield('admin_page')
  </div>
@endsection
@section('js')
  <script src="{{ {url('/')} } }/js/admin_panel.js"></script>
@show

```

components/admin_menu

```

<div id="admin_left">
  <div id="admin_left_holder">

```

```

<table class="table table-hover table-dark">
  <tr><td><a href="{{ route('admin.manage_users') }}">Manage users</a></td></tr>
  <tr><td><a href="{{ route('admin.memberships') }}">Memberships</a></td></tr>
  <tr><td><a href="{{ route('admin.surveys') }}">Surveys</a></td></tr>
  <tr><td><a href="{{ route('admin.articles') }}">Articles</a></td></tr>
  <tr><td><a href="{{ route('admin.showcase') }}">Showcase</a></td></tr>
</table>
</div>
</div>

```

components/articles

```

<section id="articles">
  @foreach($Articles as $article)
  <article>
    <h3>{{ $article->header }}</h3>
    alt }}">
    <p>{{ $article->text }}</p>
    <a class="btn btn-primary" href="{{ url('/article') }}/{{ $article->id }}">Read more</a>
  </article>
  @endforeach
</section>
<div id="articlePager">
  <button class="btn btn-primary float-left articleBtn articleBtnLeft"></button>
  <p class="text-white" id="articlePageNumber">page:1</p>
  <button class="btn btn-primary float-right articleBtn articleBtnRight"></button>
</div>
<input type="hidden" name="_token" value="{{ csrf_token() }}" id="pagerToken">

```

components/autor

```

<div class="autor">
  <div class="autor_left">

    <h3>CONTACT</h3>
    <p>ciricbgd@gmail.com</p>
    <p>8 January 1996</p>
    <p>Belgrade</p>

    <h3>LANGUAGES</h3>

    <p>Serbian</p>
    <div class="loadbar">
      <div class="loader_serbian"></div>
    </div>
    <p>English</p>
    <div class="loadbar">
      <div class="loader_english"></div>
    </div>

    <h3>INTERESTS</h3>

```

<p>Photography</p>
<p>3D Modeling:</p>
<p>Organic modeling and texturing Certificate of Completion by</p>
<p>1.The Autodesk® Authorized Training Center (ATC®)</p>
<p>2.Crater VFX Training Center</p>

</div>

<div class="autor_right">

<hgroup>
 <h1>Uroš Ćirić</h1>
 <h2>Full Stack Web Developer</h2>
</hgroup>

<h3>SKILLS</h3>

<table>
 <tr>
 <td>
 <p>HTML</p>
 <p>CSS, Sass, Bootstrap</p>
 <p>JavaScript, JQuery, AJAX, JSON</p>
 <p>Responsive design</p>
 <p>PHP, Laravel</p>
 <p>SQL</p>
 </td>
 <td>
 <p>Adobe Photoshop</p>
 <p>Adobe Lightroom</p>
 <p>Adobe Illustrator</p>
 <p>Adobe Indesign</p>
 <p>Adobe Premiere Pro</p>
 <p>Sony Vegas Pro</p>
 </td>
 </tr>
</table>
<p>Planning and looking forward to learn:</p>
<p>React</p>
<p>Node</p>

<h3>RESPONSIBILITIES</h3>

<p>Write efficient code that is easy to read and navigate as well as decrease redundancy</p>
<p>Optimize media and other content for better website performance and reduced loading time</p>
<p>UX/UI design. Create interface icons and navigation from scratch.</p>
<p>Open to learning new software and technologies</p>

<h3>Fully completed and published websites</h3>
<p>http://kaloterm.com/ - November 2017</p>

<p>Featuring admin panel, allowing owners to control the content</p>
 <p>Every element is pulled from database using php and mysql</p>

 <p>http://pogledbar.rs/ - April 2017</p>

<h3>EDUCATION</h3>
 <p>Electrical Engineering High School Nikola Tesla, Belgrade</p>
 <p>Computer Technician</p>
 <p>2011-2015</p>

 <p>ICT College of Vocational Studies, Belgrade</p>
 <p>Internet Technologies / Web Programming</p>
 <p>2015-</p>

</div>
 </div>

components/cards

```
<div class="container cards_home">
  @foreach($cards as $card)
    <div class="card">
      <div class="card_header {{ $card->color }}-gradient-color">
        <h3>{{ $card->name }}</h3>
      </div>
      <div class="card_name">${ {{ $card->price }} </div>
      <div class="card-body">
        <table class="table table-hover table-sm">
          {!! $card->perks !!}
        </table>
        <a href="#" class="btn btn-primary">Purchase</a>
      </div>
    </div>
  @endforeach
</div>

components/navbar
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand" href="{{ route($home[0]->route) }}">{{ $home[0]->name }}</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
      @if(Request::route()->getName()=='home')
        @foreach($nav as $n)
          <li class="nav-item active">
            <a class="nav-link" href="{{ $n->route }}">{{ $n->name }}<span class="sr-
only"></span></a>
          </li>
```

```

        @endforeach
    @endif
    @if($errors->any())
        <li class="text-danger">{{ $errors->first() }}</li>
    @endif
</ul>
@if(session()->has('user'))
    @if(session()->get('user')[0]->role_id > 0)
        <a href="{{ route($home[1]->route) }}" class="nav_btn"><button class="btn btn-outline-warning my-2 my-sm-0">{{ $home[1]->name }}</button></a>
    @endif
@endif
@if(session()->has('user'))
    <a href="{{ route($home[2]->route) }}" class="nav_btn"><button class="btn btn-outline-warning my-2 my-sm-0">{{ session()->get('user')[0]->username }} | {{ $home[2]->name }}</button></a>
@endif
@if(!session()->has('user'))
    <a href="{{ route('register') }}" class="nav_btn"><button class="btn btn-outline-warning my-2 my-sm-0">Register</button></a>
    <div class="nav_btn login_toggle">
        <button class="btn btn-outline-warning my-2 my-sm-0">Login</button>
    </div>
    <form action="{{ route('login') }}" method="POST" id="login_form" style="display:none">
        {{ csrf_field() }}
        <input type="text" class="form-control" name="tbUsername" placeholder="username">
        <input type="password" class="form-control" name="tbPassword" placeholder="password">
        <input type="submit" class="form-control clickable" value="Login">
    </form>
</div>
@endif
</nav>

```

components/showcase

```

<h1>The gym showcase</h1>
<h2>Our best moments</h2>
@foreach($showcase as $img)
    <a href="{{ url('/') }}/img/showcase/{{ $img->img }}" data-lightbox="gym" data-title="{{ $img->title }}">alt }}"></a>
@endforeach

```

components/survey

```

@if(!empty($surveys))
    @foreach($surveys as $survey)
        <div class="card survey_card">
            <form action="{{ route('survey.submit') }}" method="POST" class="form form-horizontal survey_form" data-survey-id="{{ $survey->id }}">

```

```

    {{ csrf_field() }}
    <ul class="survey">
        @foreach($questions as $question)
            @if($question->survey_id==$survey->id)
                <li>
                    <h6>Question #{{ $loop->iteration }}</h6>
                    <h4>{{ $question->name }}</h4>
                    <div class="form-check">
                        @foreach($answers as $answer)
                            @if($answer->question_id==$question->id)
                                <input class="form-check-input" type="{{ $question-
>answer_format }}" name="{{ $question->id }}" data-survey-id="{{ $survey->id }}" data-
question-id="{{ $question->id }}" value="{{ $answer->id }}">
                                    <label class="form-check-label">{{ $answer->name }}</label>
                                <br>
                            @endif
                        @endforeach
                    </div>
                </li>
            @endif
        @endforeach
        <li>
            <h4 style="width:100%;text-align:center;">Thank you for your time</h4>
            <br><br>
            <button type="button" class="btn btn-outline-primary survey_submit" data-
survey-id="{{ $survey->id }}">Submit</button>
        </li>
    </ul>
</form>
</div>
@endforeach
@endif

```

admin/articles

```

@extends('admin_panel')
@section('admin_page')
    <input type="hidden" value="{{ csrf_token() }}" id="token">
    <table class="table table-hover">
        <thead>
            <th>Title</th>
            <th>Image</th>
            <th>alt</th>
            <th>text</th>
            <th>delete</th>
        </thead>
        <tbody>
            @foreach($articles as $article)
                <tr>
                    <td>{{ $article->header }}</td>
                    <td>{{ $article->img }}</td>
                    <td>{{ $article->alt }}</td>

```

```

        <td style="max-width:100px;white-space: nowrap; overflow: hidden;text-
overflow: ellipsis; ">{{ $article->text }}</td>
        <td><button class="btn btn-danger deleteArticle" data-id="{{ $article-
>id }}">Delete article</button></td>
    </tr>
    @endforeach
</tbody>
</table>

<form action="{{ route('submitArticle') }}" method="POST" id="articleForm"
enctype="multipart/form-data">
    {{ csrf_field() }}
    <input type="text" name="header" class="form-control" placeholder="New article
title">
    <br>
    <input type="file" name="image" class="form-control">
    <br>
    <input type="text" name="alt" class="form-control" placeholder="Alt for image">
    <br>
    <textarea name="text" class="form-control" placeholder="Text"></textarea>
    <br>
    <input type="submit" class="btn btn-primary mb-2">
</form>
@endsection

```

admin/manage_users

```

@extends('admin_panel')
@section('admin_page')
    <table class="table table-hover table-responsive">
        <thead>
            <tr>
                <th>Firstname</th>
                <th>Lastname</th>
                <th>Username</th>
                <th>Email</th>
                <th>Birth date</th>
                <th>Phone number</th>
                <th>Role</th>
                @if(session()->get('user')[0]->role_id==3)<th>Change Role</th>@endif
                <th>edit</th>
                <th>delete</th>
            </tr>
        </thead>
        <tbody>
            @foreach($users as $user)
                <tr data-uid="{{ $user->id }}">
                    <input type="hidden" name="_token" class="token" value="{{ csrf_token() }}"
data-uid="{{ $user->id }}">
                    <td>{{ $user->firstname }}</td>
                    <td>{{ $user->lastname }}</td>
                    <td>{{ $user->username }}</td>

```



```

<td>{{ $user->email }}</td>
<td>{{ $user->date }}</td>
<td>{{ $user->phone }}</td>
<td>{{ $user->role }}</td>
@if(session()->get('user')[0]->role_id>=3)
    <td>
        <select class="form-control selectRole" data-uid="{{ $user->id }}">
            @foreach($roles as $role)
                @if($user->role!=$role->role AND $role->role!='owner')
                    <option class="form-control" value="{{ $role->id }}">{{ $role-
>role }}</option>
                @endif
            @endforeach
        </select>
    </td>
@endif
    <td><button class="btn btn-secondary btnEditUser" data-uid="{{ $user-
>id }}">Edit user</button></td>
    <td><button class="btn btn-danger btnDeleteUser" data-uid="{{ $user-
>id }}">Delete user</button></td>
</tr>
@endforeach
</tbody>
</table>
@endsection

```

admin/memberships

```

@extends('admin_panel')
@section('admin_page')
    <table class="table table-hover">
        @foreach($memberships as $membership)
            <tr>
                <td><input type="text" name="tbName" class="form-control membershipName"
value="{{ $membership->name }}" size="1" data-mid="{{ $membership->id }}"></td>
                <input type="hidden" name="_token" value="{{ csrf_token() }}" class="token"
data-mid="{{ $membership->id }}">
                <td>
                    <select name="selectColor" class="form-control membershipColor" data-
mid="{{ $membership->id }}">
                        <option value="blue" @if($membership-
>color=='blue'){{ 'selected' }}@endif>blue</option>
                        <option value="green" @if($membership-
>color=='green'){{ 'selected' }}@endif>green</option>
                        <option value="red" @if($membership->color=='red'){{ 'selected' }}
@endif>red</option>
                    </select>
                </td>
                <td>
                    <input type="text" class="form-control membershipPrice" name="tbPrice"
value="{{ $membership->price }}" size="1" maxlength="6" data-mid="{{ $membership-
>id }}">

```

```

        </div>
    </td>
    <td><button class="btn btn-secondary btnEditMembership" data-
mid="{{ $membership->id }}">Edit membership</button></td>
    <td><button class="btn btn-danger btnDeleteMembership" data-
mid="{{ $membership->id }}">Delete membership</button></td>
</tr>
@endforeach
</table>

<form action="{{ route('insertMembership') }}" method="POST" style="max-
width:320px;">
    {{ csrf_field() }}
    <input type="text" name="name" class="form-control" placeholder="New membership
name">
    <br>
    <input type="text" name="color" class="form-control" placeholder="Color">
    <br>
    <input type="text" name="price" class="form-control" placeholder="Price in $">
    <br>
    <textarea name="perks" class="form-control" placeholder="Preks&#10Seperate them
with \ slash sign"></textarea>
    <br>
    <input type="submit" class="btn btn-primary mb-2">
</form>
@endsection

```

admin/showcase

```

@extends('admin_panel')
@section('admin_page')
    <input type="hidden" name="_token" id="token" value="{{ csrf_token() }}">
    <table class="table table-hover table-responsive">
        <thead>
            <th>Title</th>
            <th>img</th>
            <th>alt</th>
            <th>delte</th>
        </thead>
        <tbody>
            @foreach($showcase as $img)
                <tr>
                    <td>{{ $img->title }}</td>
                    <td>{{ $img->img }}</td>
                    <td>{{ $img->alt }}</td>
                    <td><button class="btn btn-danger deleteShowcase" data-id="{{ $img-
>id }}">Delete showcase</button></td>
                </tr>
            @endforeach
        </tbody>
    </table>

```

```

<form action="{{url('/')}}/admin/insertShowcase" enctype="multipart/form-data"
method="POST" style="max-width:320px;">
    {{ csrf_field() }}
    <input type="text" class="form-control" name="title" placeholder="New showcase
title">
    <br>
    <input type="file" name="image" class="form-control">
    <br>
    <input type="text" class="form-control" name="alt" placeholder="image alt">
    <br>
    <input type="submit" class="btn btn-primary mb-2">
</form>
@endsection

```

admin/sruveyresults

```

@extends('admin_panel')
@section('admin_page')
<table class="table table-hover">
    <thead>
        <tr>
            <th>@foreach($survey_name as $name)
                {{ $name->name }}
            @endforeach</th>
        </tr>
    </thead>
    <tbody>
        @foreach($survey_questions as $question)
            <tr>
                <td>{{ $question->name }}</td>
                @foreach($survey_answers as $answer)
                    @if($answer->question_id==$question->id)
                        <td>
                            {{ $answer->name }}:
                            <?php $i=0;?>
                            @foreach($survey_results as $result)
                                @if($result->answer_id == $answer->id)
                                    <?php $i++;?>
                                @endif
                            @endforeach
                            <?php echo $i;?>
                        </td>
                    @endif
                @endforeach
            </tr>
        @endforeach
    </tbody>
</table>
@endsection

```

admin/surveys

```

@extends('admin_panel') @section('admin_page')

```

```

<div id="surveyForm">
  {{ csrf_field() }}
  <input type="text" id="surveyName" placeholder="Survey name" class="form-control">

  <div class="questionRow">
    <input type="text" class="form-control surveyQuestion" placeholder="Question">
    <select class="form-control questionOption" data-questionnumber="1">
      <option>radio</option>
      <option>checkbox</option>
    </select>
    <p class="clickable answerCtrl questionDelete">x</p>
    <p class="clickable answerCtrl questionAdd">+</p>
    <div class="answerRow" data-questionnumber="1">
      <input type="text" class="form-control surveyAnswer" data-questionnumber="1"
placeholder="Possible answer">
      <p class="clickable answerCtrl answerDelete">x</p>
      <p class="clickable answerCtrl answerAdd">+</p>
    </div>
    <div class="answerRow" data-questionnumber="1">
      <input type="text" class="form-control surveyAnswer" data-questionnumber="1"
placeholder="Possible answer">
      <p class="clickable answerCtrl answerDelete">x</p>
      <p class="clickable answerCtrl answerAdd">+</p>
    </div>
  </div>

  <button role="button" class="btn btn-success" id="makeSurvey">Create survey</button>
</div>

```

```

<table class="table table-hover" style="margin-top:50px;">
  @foreach($surveys as $survey)
    <tr>
      <td>{{ $survey->name }}</td>
      <td><a href="{{ url('/admin/surveys/view/'.$survey->id) }}" role="button" class="btn
btn-primary text-light">View results</a></td>
      <td><a href="{{ url('/admin/surveys/delete/'.$survey->id) }}" role="button" class="btn
btn-danger text-light">Delete survey</a></td>
    </tr>
  @endforeach
</table>
@endsection

```

