*Készítette: Györkei Lidia, project neve:* wp3gyorkei2-app

**Webprogramozás III. Második Feladathoz magyarázat**

**Project működése röviden:**

**Registerre kattintva lehet regisztrálni, Loginra kattintva belépni. Két usert létrehoztam, velük bejelentkezve ki lehet próbálni a dolgokat. Csak az admin jogosultsággal rendelkező látja a táblázatokat (adatokat, blogokat, stb), a sima user jelenleg szinte semmit sem lát, csak regisztrálni tud és bejelentkezni egy üres honlapra. Új usert is lehet regisztrálni a honlap kipróbálásához, de fontos, hogy akkor a phpmyadminban a usertype-ot át kell állitani adminra, a teljes jogosultságért:**

* **admin:** 
  + **email: admin@gmail.com**
  + **jelszó: 12345678**
* **sima user:**
  + **email: teszt.elek@gmail.com**
  + **jelszó: 12345678**

**Új diák hozzáadása (adminos bejelentkezés után katt bal felső sarokban a nyílra, majd katt „Hozzáadás 3 táblázathoz, Új diák hozzáadása”)**

* **Meg kell adni az iskola ID számát (5 vagy 6), és a Ház ID számát (1, 2, 4, 5 vagy 6). Sajnos nem tudtam megoldani, hogy ID szám helyett az iskola vagy a ház nevét is elfogadja a rendszer.**

**Adatbázis adatok .env-ben:**

* **DB\_DATABASE = wp3feladat2**
* **DB\_USERNAME = wp3feladat2**
* **jelszó: jJlHvG85Ve.rDig)**

**A munkamenet alapú hitelesítés elkészítése:**

* **Laravel/Breeze-zel (youtube tutorial alapján).**
* **Két külön jogosultság: sima userekek és admin (regisztráció után, a usertype alapvetően ’user’. Ezt a phpmyadminban lehet átállítani a usertype-nál ’admin’-ra.**

**WEB.PHP (teljes web.php kód a 8. oldalon ):**

**Kezdő oldal:**

Route::get('/', [HomeController::class, 'homepage']);  // function name: homepage

**Ha loginra kattintunk, indexre megyünk, majd bejelentkezés után a home-ra:**

Route::get('/home', [HomeController::class, 'index'])->middleware('auth')->name('home');

**HomeController: sima usereket és admint bejelentkezés után más-más oldalra irányítom:**

class HomeController extends Controller

{

    public function index(){

        if(Auth::id()){

            $usertype = Auth()->user()->usertype;

            if($usertype=='user'){ //sima usereket a resources/views/dashboard.blade.php-ra irányítom

                return view('home.homepage');

            }

            else if($usertype=='admin'){

                return view('admin.adminhome'); //admin-t pedig ide irányítom

            }

            else{

                return redirect()->back();

            }

        }

    }

    public function homepage(){

        return view('home.homepage');

    }

}

**RESOURCES/VIEWS/ADMIN mappába helyezett views-ok (nem sorolom itt fel az összeset), zárójelben pedig a honlapon szereplő elnevezések:**

* **harom\_tablazat (három táblázat egyben):** itt jelenítem meg a három összekapcsolt táblázat adatait együtt (houses, schools, students)
* **house\_page (Hozzáadás 3 táblázathoz, majd Új Ház hozzáadása):** itt tudok új házat hozzá adni a houses táblázathoz
* **show\_house:** houses táblázat adatainak megjelenítése, itt tudok törölni közülük
* **update\_house (Ház read delete, majd Javitás):** houses táblázat update lehetőség
* **school\_page (Hozzáadás 3 táblázathoz, majd Új Iskola Hozzáadása):** itt tudok új házat hozzáadni a schools táblázathoz
* **show\_school (Iskola read delete):** schools táblázat adatainak megjelenítése, itt tudok törölni közülük
* **update\_schools (Ház read delete, majd Javítás):** schools táblázat update lehetőség
* **student\_page (Hozzáadás 3 táblázathoz, majd Új diák hozzáadása):** itt tudok új házat hozzá adni a students táblázathoz
* **show\_student (Diák read delete(:** students táblázat adatainak megjelenítése, itt tudok törölni közülük
* **update\_student (Diák read delete-re katt, majd ott a javításra):** students táblázat update lehetőség
* **post\_page (Új Poszt):** új post feltöltése, FOTÓT IS ITT TUDOK FELTÖLTENI
* **show\_post (Post read delete):** összes post megtekintése, FOTÓ LETÖLTÉSE, törlés

**HOMECONTROLLER: Minden táblázatnak (users, blogs, schools, students, houses) itt valósítottam meg a CRUD műveleteit**

<?php

namespace App\Http\Controllers;

use App\Models\School;

use App\Models\Student;

use App\Models\House;

use App\Models\Post;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth; //ellenőrzi,a user logged in-e

class AdminController extends Controller

{

    public function post\_page(){

        return view('admin.post\_page');

    }

    public function add\_post(Request $request){

        $user=Auth()->user();  //nyíl után phpadminos user adatok

        $userid = $user->id;

        $name = $user->name;

        $usertype = $user->usertype;

        $post = new Post();

        $post->title = $request ->title; //első title phpmyadminból van, második viszont post\_page.blade-ből!!!

        $post->description = $request ->description;

        $post->post\_status = 'active';

        $post->user\_id = $userid;  //phpadmin post adatok, összekapcsolva a useres adatokkal

        $post->name = $name;

        $post->usertype = $usertype;

        //fenti 3 sor magyarázat: első két adat post, harmadik user

        ///////////elso 3 image kezdetű sor arra kell, hogy az image a public folderbe legyen

        $image=$request->image;

        if($image){  //if azért kell, hogy kép nélkül is postolhassunk

             $imagename=time().'.'.$image->getClientOriginalExtension();

             $request->image->move('postimage', $imagename); //az image-t a public folderben levő postimage folderbe küldjük

             /////////lenti 1 sor pedig azért, hogy az image a weboldalra kerüljön

             $post->image = $imagename;

        }

        $post->save(); //if nincs image, akkor jöhet a post és save

        return redirect()->back()->with('message', 'Sikeresen feltöltötted az új Post-ot');

    }//32. sorban, a postimage mappát magától generálja a laravel, csupán attól, hogy odairtuk és a képet is egyből oda mentette le!

    public function show\_post(){

        $post = Post::all();

        return view('admin.show\_post', compact('post'));

    }

    public function delete\_post($id){

        $post = Post::find($id);

        $post -> delete();

        return redirect()->back()->with('message', 'Sikeresen Törölted a Postot.');

    }

    public function update\_post($id){

        $post=Post::find($id); //kikeresi phpmyadminból az id-t

        return view('admin.update\_post', compact('post'));

    }

    public function edit\_post(Request $request, $id){

        $data = Post::find($id);

        $data->title = $request->title;

        $data->description = $request->description;

        $image = $request->image;

        if($image){

            $imagename = time().'.'.$image->getClientOriginalExtension();

            $request->image->move('postimage', $imagename);

            $data->image = $imagename;

        }

        $data->save();

        return redirect()->back()->with('message', 'Sikeres módosítás');

    }

    //image downloadhoz:

    public function download\_image($id){

        $post = Post::find($id);

        // Biztos van a poszthoz kép?

        if($post && $post->image){

            $path = public\_path('postimage/'.$post->image);

            // Fájl létének ellenőrzése

            if(file\_exists($path)){

                // Provide a download response

                return response()->download($path, $post->image);

            }

        }

        // Redirect back if the image is not found

        return redirect()->back()->with('message', 'A kép nem található!');

    }

    ///// 3 táblázat R egyben:

    public function harom\_tablazat(){

        $schools = School::with('houses.students')->get();

        return view('admin.harom\_tablazat', compact('schools'));

    }

    ///// SCHOOLS/ISKOLÁL functions (R, C, U, D)

    public function school\_page(){

        return view('admin.school\_page');

    }

    public function show\_school(){

        $school = School::all();

        return view('admin.show\_school', compact('school'));

    }

    public function add\_school(Request $request){

        $user=Auth()->user();  //nyíl után phpadminos school adatok

        $userid = $user->id;

        $name = $user->name;

        $usertype = $user->usertype;

        $school = new School();

        $school->school\_name = $request ->school\_name; //első school\_name phpmyadminból van, második viszont school\_page.blade-ből!!!

        $school->save();

        return redirect()->back()->with('message', 'Sikeresen feltöltötted az új iskola nevet');

    }

    public function delete\_school($id){

        $school = School::find($id);

        $school -> delete();

        return redirect()->back()->with('message', 'Sikeresen Törölted a Postot.');

    }

    public function update\_school($id){

        $school=School::find($id); //kikeresi phpmyadminból az id-t

        return view('admin.update\_school', compact('school'));

    }

    public function edit\_school(Request $request, $id){

        $data = School::find($id);

        $data->school\_name=$request->school\_name;

        $data->save();

        return redirect()->back()->with('message', 'Sikeres módosítás');

    }

    ///Csak a HOUSE/HÁZAK CRUD műveletei

    public function house\_page(){

        return view('admin.house\_page');

    }

    public function add\_house(Request $request){

        $user=Auth()->user();  //nyíl után phpadminos user adatok

        $userid = $user->id;

        $name = $user->name;

        $usertype = $user->usertype;

        $house = new House();

        $house->house\_name = $request ->house\_name;

        $house->school\_id = $request ->school\_id; //első school\_name phpmyadminból van, második viszont school\_page.blade-ből!!!

        $house->save();

        return redirect()->back()->with('message', 'Sikeresen feltöltötted az új iskola nevet');

    }

    public function show\_house(){

        $house = House::all();

        return view('admin.show\_house', compact('house'));

    }

    public function delete\_house($id){

        $house = House::find($id);

        $house -> delete();

        return redirect()->back()->with('message', 'Sikeresen Törölted a Postot.');

    }

    public function update\_house($id){

        $house=House::find($id); //kikeresi phpmyadminból az id-t

        return view('admin.update\_house', compact('house'));

    }

    public function edit\_house(Request $request, $id){

        $data = House::find($id);

        $data->house\_name=$request->house\_name;

        $data->save();

        return redirect()->back()->with('message', 'Sikeres módosítás');

    }

    ///Csak a STUDENTS/DIÁKOK CUCCAI, azaz CRUD műveletei, stb

    public function student\_page(){

        return view('admin.student\_page');

    }

    public function add\_student(Request $request){

        $user=Auth()->user();  //nyíl után phpadminos user adatok

        $userid = $user->id;

        $name = $user->name;

        $usertype = $user->usertype;

        $student = new Student();

        $student->student\_name = $request ->student\_name;

        $student->school\_id = $request ->school\_id; //első school\_name phpmyadminból van, második viszont school\_page.blade-ből!!!

        $student->house\_id = $request ->house\_id; //első school\_name phpmyadminból van, második viszont school\_page.blade-ből!!!

        $student->save();

        return redirect()->back()->with('message', 'Sikeresen feltöltötted az új iskola nevet');

    }

    public function show\_student(){

        $student = Student::all();

        return view('admin.show\_student', compact('student'));

    }

    public function delete\_student($id){

        $student = Student::find($id);

        $student -> delete();

        return redirect()->back()->with('message', 'Sikeresen Törölted a Postot.');

    }

    public function update\_student($id){

        $student=Student::find($id); //kikeresi phpmyadminból az id-t

        return view('admin.update\_student', compact('student'));

    }

    public function edit\_student(Request $request, $id){

        $data = Student::find($id);

        $data->student\_name=$request->student\_name;

        $data->save();

        return redirect()->back()->with('message', 'Sikeres módosítás');

    }

}

**Modellek:**

**A táblázatoknak létre hoztam Modelleket is, és a students, schools, houses táblázatokat így kapcsoltam össze (Ez csak a School model, de a Student és House model is hasonló elven van):**

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

class School extends Model

{

    use HasFactory;

    public function houses()

    {

        return $this->hasMany(House::class);

    }

    public function students()

    {

        return $this->hasManyThrough(Student::class, House::class);

    }

}

**WEB.PHP:**

<?php

use App\Http\Controllers\AdminController;

use App\Http\Controllers\ProfileController;

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\HomeController;

Route::get('/', [HomeController::class, 'homepage']);

Route::get('/home', [HomeController::class, 'index'])->middleware('auth')->name('home');

//ha valaki a loginra kattint, erre az oldalra irányitja (külön az admint)

Route::middleware('auth')->group(function () {

    Route::get('/profile', [ProfileController::class, 'edit'])->name('profile.edit');

    Route::patch('/profile', [ProfileController::class, 'update'])->name('profile.update');

    Route::delete('/profile', [ProfileController::class, 'destroy'])->name('profile.destroy');

});

require \_\_DIR\_\_.'/auth.php';

Route::get('/post\_page', [AdminController::class, 'post\_page']);

//utóbbi post\_page a function név, amit AdminController.php-ba is kell irni!

Route::post('/add\_post', [AdminController::class, 'add\_post']);

Route::get('/show\_post', [AdminController::class, 'show\_post']);

Route::get('/delete\_post/{id}', [AdminController::class, 'delete\_post']);

Route::get('/update\_post/{id}', [AdminController::class, 'update\_post']);

Route::post('/edit\_post/{id}', [AdminController::class, 'edit\_post']);

Route::get('/download\_image/{id}', [AdminController::class, 'download\_image']);

////3 táblázatnak 'R'

Route::get('/school\_page', [AdminController::class, 'school\_page']);

Route::get('/student\_page', [AdminController::class, 'student\_page']);

Route::get('/house\_page', [AdminController::class, 'house\_page']);

////// 3 táblázatnak 'C'

Route::post('/add\_school', [AdminController::class, 'add\_school']);

Route::post('/add\_student', [AdminController::class, 'add\_student']);

Route::post('/add\_house', [AdminController::class, 'add\_house']);

///// 3 táblázatnak 'D'

Route::get('/delete\_school/{id}', [AdminController::class, 'delete\_school']);

Route::get('/delete\_student/{id}', [AdminController::class, 'delete\_student']);

Route::get('/delete\_house/{id}', [AdminController::class, 'delete\_house']);

////3 táblázatnak 'U'

Route::get('/update\_school/{id}', [AdminController::class, 'update\_school']);

Route::get('/update\_student/{id}', [AdminController::class, 'update\_student']);

Route::get('/update\_house/{id}', [AdminController::class, 'update\_house']);

Route::post('/edit\_school/{id}', [AdminController::class, 'edit\_school']);

Route::post('/edit\_student/{id}', [AdminController::class, 'edit\_student']);

Route::post('/edit\_house/{id}', [AdminController::class, 'edit\_house']);

/////3 táblázat egyben:

Route::get('/harom\_tablazat', [AdminController::class, 'harom\_tablazat'])->name('harom\_tablazat');

Route::get('/show\_student', [AdminController::class, 'show\_student']);

Route::get('/show\_school', [AdminController::class, 'show\_school']);

Route::get('/show\_house', [AdminController::class, 'show\_house']);Route::post('/edit\_house/{id}', [AdminController::class, 'edit\_house']);

Az elkészítéséhez az alábbi tutorial videókat is felhasználtam, css-hez pedig sablont töltötöttem le:

<https://github.com/yaminshakil/Blog_Template>

<https://www.youtube.com/watch?v=veBSb9JiKTA&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=2>

<https://www.youtube.com/watch?v=qJZyvfcUzEE&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=4>

<https://www.youtube.com/watch?v=qJZyvfcUzEE&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=4>

<https://www.youtube.com/watch?v=tFHa-B45vkI&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=5>

<https://www.youtube.com/watch?v=iaP1hMUd2r0&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=6>

<https://www.youtube.com/watch?v=2mshd24KN4s&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=7>

<https://www.youtube.com/watch?v=poJYEeIGtcA&list=PLm8sgxwSZoffiTghh0cqMj0tGlonrouWX&index=9>