**Multi Auth System in Laravel**

**Create a project**

**Install laravel breeze**

Composer require laravel/breeze

Php artisan breeze:install

Npm install

Npm run dev

public function up()

    {

        Schema::create('users', function (Blueprint $table) {

            $table->id();

            $table->string('fname')->nullable();

            $table->string('lname')->nullable();

            $table->string('address')->nullable();

            $table->string('phone')->nullable();

            $table->string('email')->unique();

            $table->timestamp('email\_verified\_at')->nullable();

            $table->string('ref\_num')->nullable();

            $table->string('password');

            $table->rememberToken();

            $table->timestamps();

        });

    }

Php artisan migrate

Create another model for admin

Php artisan make:model admin –m

Make change to table migration

public function up()

    {

        Schema::create('admins', function (Blueprint $table) {

            $table->id();

            $table->string('fname')->nullable();

            $table->string('lname')->nullable();

            $table->string('address')->nullable();

            $table->string('phone')->nullable();

            $table->string('email')->unique();

            $table->timestamp('email\_verified\_at')->nullable();

            $table->string('ref\_num')->nullable();

            $table->string('password');

            $table->rememberToken();

            $table->timestamps();

        });

    }

Make changes to model. Copy complete user model and paste in admin model and change name to admin from user as we pasted code of user model in admin model

And write this

Protected $guard = ‘admin’

// this means we will be protected its route from admin

Complete code

<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Foundation\Auth\User as Authenticatable;

use Illuminate\Notifications\Notifiable;

use Laravel\Sanctum\HasApiTokens;

class admin extends Authenticatable

{

    protected $guard = 'admin';

    use HasApiTokens, HasFactory, Notifiable;

    /\*\*

     \* The attributes that are mass assignable.

     \*

     \* @var array<int, string>

     \*/

    protected $fillable = [

        'name',

        'email',

        'password',

    ];

    /\*\*

     \* The attributes that should be hidden for serialization.

     \*

     \* @var array<int, string>

     \*/

    protected $hidden = [

        'password',

        'remember\_token',

    ];

    /\*\*

     \* The attributes that should be cast.

     \*

     \* @var array<string, string>

     \*/

    protected $casts = [

        'email\_verified\_at' => 'datetime',

    ];

}

Now main part

Go to your config/auth.php

 'guards' => [

        'web' => [

            'driver' => 'session',

            'provider' => 'users',

        ],

    ],

After web make admin guard

 'guards' => [

        'web' => [

            'driver' => 'session',

            'provider' => 'users',

        ],

        'admin' => [

            'driver' => 'session',

            'provider' => 'admin',

        ],

    ],

Now we see after provider we have admin written. Let’s go below and make provide of admin as well. First let’s show original code

   'providers' => [

        'users' => [

            'driver' => 'eloquent',

            'model' => App\Models\User::class,

        ],

        // 'users' => [

        //     'driver' => 'database',

        //     'table' => 'users',

        // ],

    ],

Here replace users with admin

 'providers' => [

        'users' => [

            'driver' => 'eloquent',

            'model' => App\Models\User::class,

        ],

        'admins' => [

            'driver' => 'eloquent',

            'model' => App\Models\admin::class,

        ],

        // 'users' => [

        //     'driver' => 'database',

        //     'table' => 'users',

        // ],

    ],

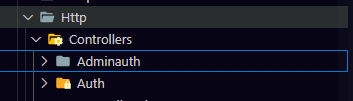
We can migrate now

Php artisan migrate

**Important**

Now we have app/http/controller/auth

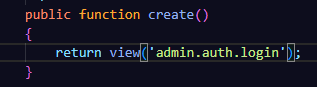
Duplicate auth folder with name Adminauht



Now change namespace to Adminauth in all files inside the Adminauth folder.

namespace App\Http\Controllers\Adminauth;

now we have a file in both auth or Adminauth/authenticatesession let’s understandthis



If we want different view we need to define like this for admin

 public function store(LoginRequest $request)

    {

        $request->authenticate();

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

        return redirect()->intended(RouteServiceProvider::Admin\_home);

    }

We change HOME to Admin\_home for redirect after admin login

Let’s go to routeserviceprovider.php

 public const HOME = '/dashboard';

    public const Admin\_home = '/admin/dashboard';

so we have define admin/dashboard redirect after admin login and dashboard route for user login

 public function destroy(Request $request)

    {

        Auth::guard('admin')->logout();

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

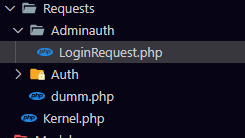
        $request->session()->regenerateToken();

        return redirect('/');

    }

So we change the destroy guard for admin as well

Now lets duplicate this folder **App/Http/Requests/Admin/Auth** to **Adminauth and change the namespace as well**

****

<?php

namespace App\Http\Requests\Adminauth;

use Illuminate\Auth\Events\Lockout;

use Illuminate\Foundation\Http\FormRequest;

use Illuminate\Support\Facades\Auth;

use Illuminate\Support\Facades\RateLimiter;

use Illuminate\Support\Str;

use Illuminate\Validation\ValidationException;

class LoginRequest extends FormRequest

{

**In the same file we modify authenticatefunction**

 public function authenticate()

    {

*$this*->ensureIsNotRateLimited();

        if (! Auth::guard('admin')->attempt(*$this*->only('email', 'password'), *$this*->boolean('remember'))) {

            RateLimiter::hit(*$this*->throttleKey());

            throw ValidationException::withMessages([

                'email' => trans('auth.failed'),

            ]);

        }

        RateLimiter::clear(*$this*->throttleKey());

    }

**We have use admin guard** So this will login logout from admin guard models

**IMPORTANT**

Now go to **App\Htpp\Controllers\Adminauth\Authenticatesessioncontroller.php**

**Modify**

namespace App\Http\Controllers\Adminauth;

use App\Http\Controllers\Controller;

use App\Http\Requests\Adminauth\LoginRequest;

use App\Providers\RouteServiceProvider;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth;

**here we have modify namespace for LoginRequest**

**IMPORTANT**

we need to go to **App\Htpp\Controllers\Adminauth\RegisterController.php**

make few changes which are listed below

public function create()

    {

        return view('admin.auth.register');

    }

     $request->validate([

            'name' => ['required', 'string', 'max:255'],

            'email' => ['required', 'string', 'email', 'max:255', 'unique:admins'],

            'password' => ['required', 'confirmed', Rules\Password::defaults()],

        ]);

    $user = admin::create([

            'name' => $request->name,

            'email' => $request->email,

            'password' => Hash::make($request->password),

        ]);

   Auth::guard('admin')->login($user);

   event(new Registered($user));

        Auth::guard('admin')->login($user);

// redirect to Admin\_home

        return redirect(RouteServiceProvider::Admin\_home);

    }

Complete store function after modification

 public function store(Request $request)

    {

        $request->validate([

            'name' => ['required', 'string', 'max:255'],

            'email' => ['required', 'string', 'email', 'max:255', 'unique:admins'],

            'password' => ['required', 'confirmed', Rules\Password::defaults()],

        ]);

        $user = admin::create([

            'name' => $request->name,

            'email' => $request->email,

            'password' => Hash::make($request->password),

        ]);

        event(new Registered($user));

        Auth::guard('admin')->login($user);

        return redirect(RouteServiceProvider::Admin\_home);

    }

**WEB.PHP**

Route::get('/dashboard', function () {

    return view('dashboard');

})->middleware(['auth'])->name('dashboard');

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

//we just duplicated above code. Let’s make modifications

Route::get('/dashboard', function () {

    return view('dashboard');

})->middleware(['auth'])->name('dashboard');

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

Route::get('/admin/dashboard', function () {

    return view('admin.dashboard');

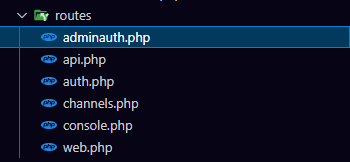
})->middleware(['auth:admin'])->name('admin.dashboard');

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

so we have adminauth.php

basically we have auth.php in routes/auth.php

we have to duplicate this.



Like this

use App\Http\Controllers\Auth\AuthenticatedSessionController;

use App\Http\Controllers\Auth\ConfirmablePasswordController;

use App\Http\Controllers\Auth\EmailVerificationNotificationController;

use App\Http\Controllers\Auth\EmailVerificationPromptController;

use App\Http\Controllers\Auth\NewPasswordController;

use App\Http\Controllers\Auth\PasswordResetLinkController;

use App\Http\Controllers\Auth\RegisteredUserController;

use App\Http\Controllers\Auth\VerifyEmailController;

use Illuminate\Support\Facades\Route;

replace auth to Adminauth(this is the folder we duplicated and managed above)

<?php

use App\Http\Controllers\Auth\AuthenticatedSessionController;

use App\Http\Controllers\Auth\ConfirmablePasswordController;

use App\Http\Controllers\Auth\EmailVerificationNotificationController;

use App\Http\Controllers\Auth\EmailVerificationPromptController;

use App\Http\Controllers\Auth\NewPasswordController;

use App\Http\Controllers\Auth\PasswordResetLinkController;

use App\Http\Controllers\Auth\RegisteredUserController;

use App\Http\Controllers\Auth\VerifyEmailController;

use Illuminate\Support\Facades\Route;

//major change

Route::group(['middleware' => ['guest:admin'] , 'prefix' => 'admin' , 'as' => 'admin.'] ,function(){

    Route::get('register', [RegisteredUserController::class, 'create'])

                ->name('register');

    Route::post('register', [RegisteredUserController::class, 'store']);

    Route::get('login', [AuthenticatedSessionController::class, 'create'])

                ->name('login');

    Route::post('login', [AuthenticatedSessionController::class, 'store']);

    Route::get('forgot-password', [PasswordResetLinkController::class, 'create'])

                ->name('password.request');

    Route::post('forgot-password', [PasswordResetLinkController::class, 'store'])

                ->name('password.email');

    Route::get('reset-password/{token}', [NewPasswordController::class, 'create'])

                ->name('password.reset');

    Route::post('reset-password', [NewPasswordController::class, 'store'])

                ->name('password.update');

});

Route::group(['middleware' => ['guest:admin'] , 'prefix' => 'admin' , 'as' => 'admin.'] ,function(){

    Route::get('verify-email', [EmailVerificationPromptController::class, '\_\_invoke'])

                ->name('verification.notice');

    Route::get('verify-email/{id}/{hash}', [VerifyEmailController::class, '\_\_invoke'])

                ->middleware(['signed', 'throttle:6,1'])

                ->name('verification.verify');

    Route::post('email/verification-notification', [EmailVerificationNotificationController::class, 'store'])

                ->middleware('throttle:6,1')

                ->name('verification.send');

    Route::get('confirm-password', [ConfirmablePasswordController::class, 'show'])

                ->name('password.confirm');

    Route::post('confirm-password', [ConfirmablePasswordController::class, 'store']);

    Route::post('logout', [AuthenticatedSessionController::class, 'destroy'])

                ->name('logout');

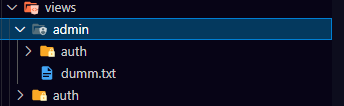
});

**IMPORTANT**

**Let’s maintain the views**

**Resources/views/auth copy this folder inside the admin/auth**

**Now we have these**

****

**Inside admin/auth/login.php** do these changes now

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

            @csrf

            <div class="flex items-center justify-end mt-4">

                @if (Route::has('password.request'))

                    <a class="underline text-sm text-gray-600 hover:text-gray-900" href="{{ route('admin.password.request') }}">

                        {{ \_\_('Forgot your password?') }}

                    </a>

                @endif

                <x-button class="ml-3">

                    {{ \_\_('Log in') }}

                </x-button>

            </div>

        <h4 class="text-4xl font-bold text-center">AdminLogin</h4>

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

            @csrf

Same things for register

 <form method="POST" action="{{ route('admin.register') }}">

            @csrf

  <div class="flex items-center justify-end mt-4">

                <a class="underline text-sm text-gray-600 hover:text-gray-900" href="{{ route('admin.login') }}">

                    {{ \_\_('Already registered?') }}

                </a>

                <x-button class="ml-4">

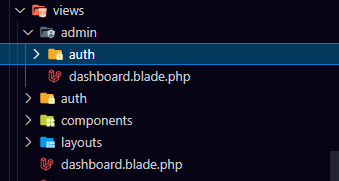
                    {{ \_\_('Register') }}

                </x-button>

            </div>

**IMPORTANT**

**Let’s paste onedashboard file inside admin/**

****

**Let’s make separate layout for our dashboard**

**So copy layout folder from resources/views and paste to resources/views/admin/**

So when we open file **resources/views/admin/layouts/app.blade.php**

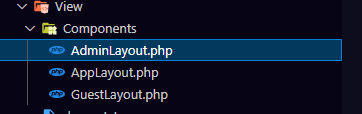
So we will find @include

            @include('layouts.navigation')

Let’s change this to admin.layouts.navigation

            @include('admin.layouts.navigation')

App/view/component/ copy AppLayout.php to AdminLayout.php



Change the class name as well

<?php

namespace App\View\Components;

use Illuminate\View\Component;

class AdminLayout extends Component

{

    /\*\*

     \* Get the view / contents that represents the component.

     \*

     \* @return \Illuminate\View\View

     \*/

    public function render()

    {

    public function render()

    {

        return view('admin.layouts.app');

    }

Complete change

<?php

namespace App\View\Components;

use Illuminate\View\Component;

class AdminLayout extends Component

{

    /\*\*

     \* Get the view / contents that represents the component.

     \*

     \* @return \Illuminate\View\View

     \*/

    public function render()

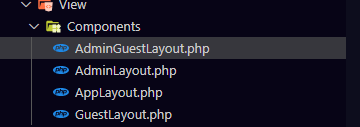
    {

        return view('admin.layouts.app');

    }

}

Do the same for the guest



<?php

namespace App\View\Components;

use Illuminate\View\Component;

class AdminGuestLayout extends Component

{

    /\*\*

     \* Get the view / contents that represents the component.

     \*

     \* @return \Illuminate\View\View

     \*/

    public function render()

    {

        return view('admin.layouts.guest');

    }

}

Now inside resources/vies/admin

Open dashaboard.blade.php

Use x-admin-layout

<x-admin-layout>

</x-admin-layout>

Now for login/register inside views/admin/auth

Use this

<x-admin-guest-layout>

</x-admin-guest-layout>

Perfect

Let’s make adjustment to welcome.blade.php