# lyhienminh87@gmail.comMultiple Authentication System Laravel 5.4 | 5.5

At first install a fresh 5.4 laravel project .

1.Create a admins migration table

<?php  
use Illuminate\Support\Facades\Schema;  
use Illuminate\Database\Schema\Blueprint;  
use Illuminate\Database\Migrations\Migration;  
class CreateAdminsTable extends Migration  
{  
 /\*\*  
 \* Run the migrations.  
 \*  
 \* @return void  
 \*/  
 public function up()  
 {  
 Schema::create('admins', function (Blueprint $table) {  
 $table->increments('id');  
 $table->string('name');  
 $table->string('email')->unique();  
 $table->string('job\_title');  
 $table->string('password');  
 $table->rememberToken();  
 $table->timestamps();  
 });  
 }  
 /\*\*  
 \* Reverse the migrations.  
 \*  
 \* @return void  
 \*/  
 public function down()  
 {  
 Schema::dropIfExists('admins');  
 }  
}

2.Admin Model

<?php  
  
namespace App;  
  
use Illuminate\Notifications\Notifiable;  
use Illuminate\Foundation\Auth\User as Authenticatable;  
  
class Admin extends Authenticatable  
{  
 use Notifiable;  
  
 protected $guard = 'admin';  
  
 /\*\*  
 \* The attributes that are mass assignable.  
 \*  
 \* @var array  
 \*/  
 protected $fillable = [  
 'name', 'email', 'password', 'job\_title',  
 ];  
  
 /\*\*  
 \* The attributes that should be hidden for arrays.  
 \*  
 \* @var array  
 \*/  
 protected $hidden = [  
 'password', 'remember\_token',  
 ];  
}

3.go to App/Exceptions/Handler.php >>> then paste all code

<?php  
namespace App\Exceptions;  
use Exception;  
use Illuminate\Auth\AuthenticationException;  
use Illuminate\Foundation\Exceptions\Handler as ExceptionHandler;  
class Handler extends ExceptionHandler  
{  
 /\*\*  
 \* A list of the exception types that should not be reported.  
 \*  
 \* @var array  
 \*/  
 protected $dontReport = [  
 \Illuminate\Auth\AuthenticationException::class,  
 \Illuminate\Auth\Access\AuthorizationException::class,  
 \Symfony\Component\HttpKernel\Exception\HttpException::class,  
 \Illuminate\Database\Eloquent\ModelNotFoundException::class,  
 \Illuminate\Session\TokenMismatchException::class,  
 \Illuminate\Validation\ValidationException::class,  
 ];  
 /\*\*  
 \* Report or log an exception.  
 \*  
 \* This is a great spot to send exceptions to Sentry, Bugsnag, etc.  
 \*  
 \* @param \Exception $exception  
 \* @return void  
 \*/  
 public function report(Exception $exception)  
 {  
 parent::report($exception);  
 }  
 /\*\*  
 \* Render an exception into an HTTP response.  
 \*  
 \* @param \Illuminate\Http\Request $request  
 \* @param \Exception $exception  
 \* @return \Illuminate\Http\Response  
 \*/  
 public function render($request, Exception $exception)  
 {  
 return parent::render($request, $exception);  
 }  
 /\*\*  
 \* Convert an authentication exception into an unauthenticated response.  
 \*  
 \* @param \Illuminate\Http\Request $request  
 \* @param \Illuminate\Auth\AuthenticationException $exception  
 \* @return \Illuminate\Http\Response  
 \*/  
 protected function unauthenticated($request, AuthenticationException $exception)  
 {  
 if ($request->expectsJson()) {  
 return response()->json(['error' => 'Unauthenticated.'], 401);  
 }  
 $guard = array\_get($exception->guards(), 0);  
 switch ($guard) {  
 case 'admin':  
 $login = 'admin.login';  
 break;  
 default:  
 $login = 'login';  
 break;  
 }  
 return redirect()->guest(route($login));  
 }  
}

4. **go to app/Http/Middleware/RedirectIfAuthenticated.php and paste code:**

<?php  
namespace App\Http\Middleware;  
use Closure;  
use Illuminate\Support\Facades\Auth;  
class RedirectIfAuthenticated  
{  
 /\*\*  
 \* Handle an incoming request.  
 \*  
 \* @param \Illuminate\Http\Request $request  
 \* @param \Closure $next  
 \* @param string|null $guard  
 \* @return mixed  
 \*/  
 public function handle($request, Closure $next, $guard = null)  
 {  
 switch ($guard) {  
 case 'admin':  
 if (Auth::guard($guard)->check()) {  
 return redirect()->route('admin.dashboard');  
 }  
 break;  
 default:  
 if (Auth::guard($guard)->check()) {  
 return redirect('/home');  
 }  
 break;  
 }  
 return $next($request);  
 }  
}

5. Create Admin controller

<?php  
namespace App\Http\Controllers;  
use Illuminate\Http\Request;  
class AdminController extends Controller  
{  
 /\*\*  
 \* Create a new controller instance.  
 \*  
 \* @return void  
 \*/  
 public function \_\_construct()  
 {  
 $this->middleware('auth:admin');  
 }  
 /\*\*  
 \* Show the application dashboard.  
 \*  
 \* @return \Illuminate\Http\Response  
 \*/  
 public function index()  
 {  
 return view('admin');  
 }  
}

6.php artisan make:controller auth/AdminLoginController

or create AdminLoginController this app/Http/Controllers/Auth/AdminLoginController.php

and paste code:

<?php  
namespace App\Http\Controllers\Auth;  
use Illuminate\Http\Request;  
use App\Http\Controllers\Controller;  
use Auth;  
class AdminLoginController extends Controller  
{  
 public function \_\_construct()  
 {  
 $this->middleware('guest:admin');  
 }  
 public function showLoginForm()  
 {  
 return view('auth.admin-login');  
 }  
 public function login(Request $request)  
 {  
 // Validate the form data  
 $this->validate($request, [  
 'email' => 'required|email',  
 'password' => 'required|min:6'  
 ]);  
 // Attempt to log the user in  
 if (Auth::guard('admin')->attempt(['email' => $request->email, 'password' => $request->password], $request->remember)) {  
 // if successful, then redirect to their intended location  
 return redirect()->intended(route('admin.dashboard'));  
 }  
 // if unsuccessful, then redirect back to the login with the form data  
 return redirect()->back()->withInput($request->only('email', 'remember'));  
 }  
}

7. go to **config/auth.php and paste**

<?php  
return [  
 /\*  
 |--------------------------------------------------------------------------  
 | Authentication Defaults  
 |--------------------------------------------------------------------------  
 |  
 | This option controls the default authentication "guard" and password  
 | reset options for your application. You may change these defaults  
 | as required, but they're a perfect start for most applications.  
 |  
 \*/  
 'defaults' => [  
 'guard' => 'web',  
 'passwords' => 'users',  
 ],  
 /\*  
 |--------------------------------------------------------------------------  
 | Authentication Guards  
 |--------------------------------------------------------------------------  
 |  
 | Next, you may define every authentication guard for your application.  
 | Of course, a great default configuration has been defined for you  
 | here which uses session storage and the Eloquent user provider.  
 |  
 | All authentication drivers have a user provider. This defines how the  
 | users are actually retrieved out of your database or other storage  
 | mechanisms used by this application to persist your user's data.  
 |  
 | Supported: "session", "token"  
 |  
 \*/  
 'guards' => [  
 'web' => [  
 'driver' => 'session',  
 'provider' => 'users',  
 ],  
 'api' => [  
 'driver' => 'token',  
 'provider' => 'users',  
 ],  
 'admin' => [  
 'driver' => 'session',  
 'provider' => 'admins',  
 ],  
 'admin-api' => [  
 'driver' => 'token',  
 'provider' => 'admins',  
 ],  
 ],  
 /\*  
 |--------------------------------------------------------------------------  
 | User Providers  
 |--------------------------------------------------------------------------  
 |  
 | All authentication drivers have a user provider. This defines how the  
 | users are actually retrieved out of your database or other storage  
 | mechanisms used by this application to persist your user's data.  
 |  
 | If you have multiple user tables or models you may configure multiple  
 | sources which represent each model / table. These sources may then  
 | be assigned to any extra authentication guards you have defined.  
 |  
 | Supported: "database", "eloquent"  
 |  
 \*/  
 'providers' => [  
 'users' => [  
 'driver' => 'eloquent',  
 'model' => App\User::class,  
 ],  
 'admins' => [  
 'driver' => 'eloquent',  
 'model' => App\Admin::class,  
 ],  
 // 'users' => [  
 // 'driver' => 'database',  
 // 'table' => 'users',  
 // ],  
 ],  
 /\*  
 |--------------------------------------------------------------------------  
 | Resetting Passwords  
 |--------------------------------------------------------------------------  
 |  
 | You may specify multiple password reset configurations if you have more  
 | than one user table or model in the application and you want to have  
 | separate password reset settings based on the specific user types.  
 |  
 | The expire time is the number of minutes that the reset token should be  
 | considered valid. This security feature keeps tokens short-lived so  
 | they have less time to be guessed. You may change this as needed.  
 |  
 \*/  
 'passwords' => [  
 'users' => [  
 'provider' => 'users',  
 'table' => 'password\_resets',  
 'expire' => 60,  
 ],  
 'admins' => [  
 'provider' => 'admins',  
 'table' => 'password\_resets',  
 'expire' => 15,  
 ],  
 ],  
];

8..Route file web route

<?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::get('/', function () {  
 return view('welcome');  
});  
Auth::routes();  
Route::get('/home', 'HomeController@index');  
 Route::prefix('admin')->group(function() {  
 Route::get('/login', 'Auth\AdminLoginController@showLoginForm')->name('admin.login');  
 Route::post('/login', 'Auth\AdminLoginController@login')->name('admin.login.submit');  
 Route::get('/', 'AdminController@index')->name('admin.dashboard');  
 });

9. create admin.blade.php that is same home.blade.php ... paste

@extends('layouts.app')  
  
@section('content')  
<div class="container">  
 <div class="row">  
 <div class="col-md-8 col-md-offset-2">  
 <div class="panel panel-default">  
 <div class="panel-heading">ADMIN Dashboard</div>  
  
 <div class="panel-body">  
 You are logged in as <strong>ADMIN</strong>  
 </div>  
 </div>  
 </div>  
 </div>  
</div>  
@endsection

10.create views/auth/admin-login.blade.php

paste this

@extends('layouts.app')  
  
@section('content')  
<div class="container">  
 <div class="row">  
 <div class="col-md-8 col-md-offset-2">  
 <div class="panel panel-default">  
 <div class="panel-heading">ADMIN Login</div>  
 <div class="panel-body">  
 <form class="form-horizontal" role="form" method="POST" action="{{ route('admin.login.submit') }}">  
 {{ csrf\_field() }}  
  
 <div class="form-group{{ $errors->has('email') ? ' has-error' : '' }}">  
 <label for="email" class="col-md-4 control-label">E-Mail Address</label>  
  
 <div class="col-md-6">  
 <input id="email" type="email" class="form-control" name="email" value="{{ old('email') }}" required autofocus>  
  
 @if ($errors->has('email'))  
 <span class="help-block">  
 <strong>{{ $errors->first('email') }}</strong>  
 </span>  
 @endif  
 </div>  
 </div>  
  
 <div class="form-group{{ $errors->has('password') ? ' has-error' : '' }}">  
 <label for="password" class="col-md-4 control-label">Password</label>  
  
 <div class="col-md-6">  
 <input id="password" type="password" class="form-control" name="password" required>  
  
 @if ($errors->has('password'))  
 <span class="help-block">  
 <strong>{{ $errors->first('password') }}</strong>  
 </span>  
 @endif  
 </div>  
 </div>  
  
 <div class="form-group">  
 <div class="col-md-6 col-md-offset-4">  
 <div class="checkbox">  
 <label>  
 <input type="checkbox" name="remember" {{ old('remember') ? 'checked' : '' }}> Remember Me  
 </label>  
 </div>  
 </div>  
 </div>  
  
 <div class="form-group">  
 <div class="col-md-8 col-md-offset-4">  
 <button type="submit" class="btn btn-primary">  
 Login  
 </button>  
  
 <a class="btn btn-link" href="{{ route('password.request') }}">  
 Forgot Your Password?  
 </a>  
 </div>  
 </div>  
 </form>  
 </div>  
 </div>  
 </div>  
 </div>  
</div>  
@endsection

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~Complte ~~~~~~~~~~~~~~~~~~~~~~~~~~

Source /Courtesy [J. Alexander Curtis](https://medium.com/@jacurtis) .Many many thanks dear.

*5.4.\* multi-auth by*[*J. Alexander Curtis*](https://medium.com/@jacurtis)

[DevMarketer/multiauth\_tutorial  
Contribute to multiauth\_tutorial development by creating an account on GitHub.github.com](https://github.com/DevMarketer/multiauth_tutorial)

*5.5.\* multi-auth by*[*Mostafa Kamal*](https://medium.com/@code4mk)

[code4mk/multi-auth-laravel  
multi-auth-laravel - Multi-auth system laravel 5.4 , 5.5 . courtesy : @jacurtisgithub.com](https://github.com/code4mk/multi-auth-laravel)

*Stay with****Hello laravel***

[](https://www.facebook.com/sharer/sharer.php?u=https%3A//medium.com/hello-laravel/multiple-authentication-system-laravel-5-4-ac94c759638a)

[](https://twitter.com/home?status=https%3A//medium.com/hello-laravel/multiple-authentication-system-laravel-5-4-ac94c759638a)

* [Laravel](https://medium.com/tag/laravel?source=post)
* [PHP](https://medium.com/tag/php?source=post)

One clap, two clap, three clap, forty?

By clapping more or less, you can signal to us which stories really stand out.

241

20

 Follow

[](https://medium.com/@code4mk?source=footer_card)

[**Mostafa Kamal**](https://medium.com/@code4mk)

A code artist || Developer || Designer .

 Follow

[](https://medium.com/hello-laravel?source=footer_card)

[**Hello Laravel**](https://medium.com/hello-laravel?source=footer_card)

Hello Laravel is a laravel framework publication https: // hellolaravel.org

[More from Hello Laravel](https://medium.com/hello-laravel/dropbox-filesystem-laravel-5-4-dfb60a618d6e?source=placement_card_footer_grid---------0-41)

[Dropbox Filesystem Laravel 5.4+](https://medium.com/hello-laravel/dropbox-filesystem-laravel-5-4-dfb60a618d6e?source=placement_card_footer_grid---------0-41)

[Go to the profile of Mostafa Kamal](https://medium.com/@code4mk)

[Mostafa Kamal](https://medium.com/@code4mk?source=placement_card_footer_grid---------0-41)

17

a