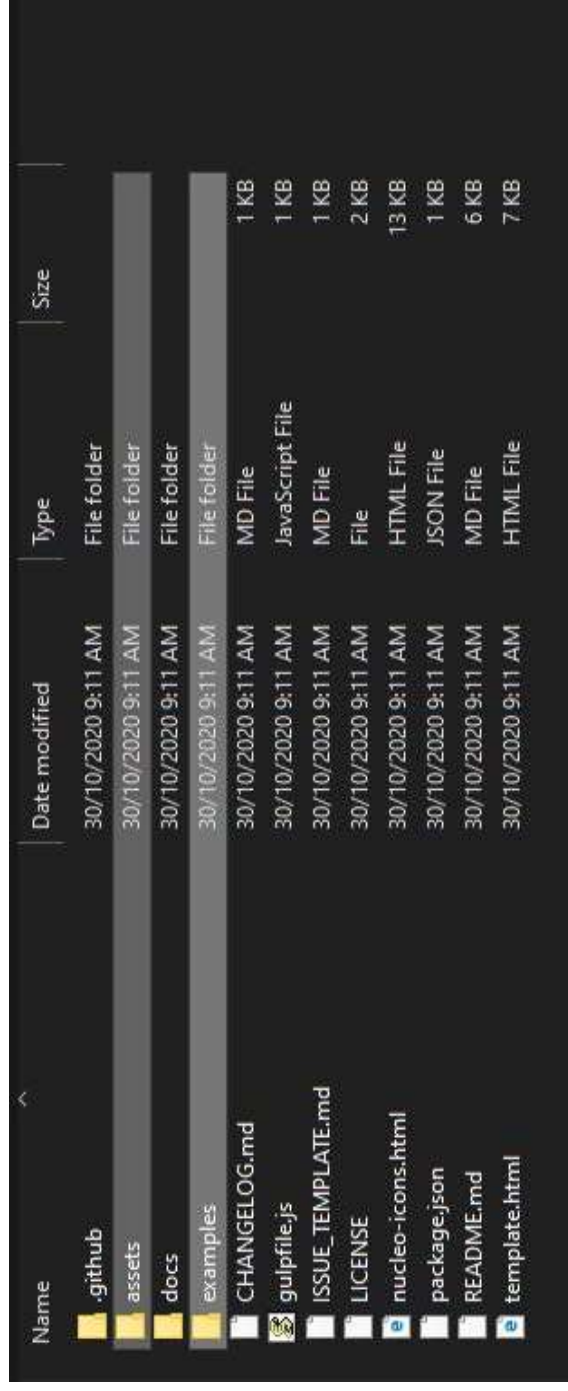


## Unit 5 -Admin Panel Setup

# Setup Admin Panel using Bootstrap theme

- First of all, download a Bootstrap Admin Template. Unzip the template file, we will be using the assets and examples folders.



Name	Date modified	Type	Size
.github	30/10/2020 9:11 AM	File folder	
assets	30/10/2020 9:11 AM	File folder	
docs	30/10/2020 9:11 AM	File folder	
examples	30/10/2020 9:11 AM	File folder	
CHANGELOG.md	30/10/2020 9:11 AM	MD File	1 KB
gulpfile.js	30/10/2020 9:11 AM	JavaScript File	1 KB
ISSUE_TEMPLATE.md	30/10/2020 9:11 AM	MD File	1 KB
LICENSE	30/10/2020 9:11 AM	File	2 KB
nucleo-icons.html	30/10/2020 9:11 AM	HTML File	13 KB
package.json	30/10/2020 9:11 AM	JSON File	1 KB
README.md	30/10/2020 9:11 AM	MD File	6 KB
template.html	30/10/2020 9:11 AM	HTML File	7 KB

- Step 1: Copy the assets folder and paste it into your Laravel web project public folder.
- Step 2: Go to resources-> views create two folder layouts and admin.
- Step 3: Create a file name dashboard.blade.php inside the admin folder.
- Step 4: Copy the html file from the examples folder, rename it to master.blade.php, and paste it into the layouts folder.

```
✓ resources
  > css
  > js
  > lang
  ✓ views
  ✓ admin
    📁 dashboard.blade.php
  ✓ layouts
    📁 master.blade.php
    📁 welcome.blade.php
```

- Step 5: Open master.blade.php file, cut the content inside `<div class="content"> ... </div>` and paste it to dashboard.blade.php. Then replace the content with `@yield('content')`
- Step 6: Open dashboard.blade.php file, insert `@section('content')` at the first line before the content, and `@endsection` at the last line.

- **master.blade.php**
- **...**
- **<div class="content">**
- **@yield('content')**
- **</div>**
- **...dashboard.blade.php**
- **@section('content')**
- **<div**  
**class="container-fluid">**
- **...**
- **</div>**
- **@endsection**

- Step 7: Open the master.blade.php, replace all the resources path as below:
- Change
  - ../assets/css/material-dashboard.css?v=2.1.2
  - To
  - {{ asset('public/assets/css/material-dashboard.css?v=2.1.2') }}
- Step 8: Go to the routes/web.php file, add the URL path for the admin
  - Route::get('/admin', function () {
  - return view('admin.dashboard');
  - });
  - Now, you can test your site by accessing the URL:
  - <http://localhost/<root-folder name>/admin>

# Creating Admin Login

- Steps to Make Admin Auth in Laravel 8:
- Step 1: Create an admin table in the database
- Create one migration for admin and add “name, email, password, created\_at, updated\_at” fields in the “admins” table
- Step 2: Add Guards
- To add guards, you need to make the following changes in the config/auth.php file.

```
1 'guards' => [
2   'web' => [
3     'driver' => 'session',
4     'provider' => 'users',
5   ],
6   'admin' => [
7     'driver' => 'session',
8     'provider' => 'admins',
9   ],
10 ],
11
12 'providers' => [
13   'users' => [
14     'driver' => 'eloquent',
15     'model' => App\Models\User::class,
16   ],
17   'admins' => [
18     'driver' => 'eloquent',
19     'model' => App\Models\Admin::class,
20   ],
21 ],
```



- Step 3: Add admin auth middleware
- Run the below command
- `php artisan make:middleware AdminAuthenticated`
- Now, move to the below path
- `app/Http/Middleware/AdminAuthenticated.php`
- And add the code as follows

```

1 <?php
2
3 namespace App\Http\Middleware;
4
5 use Closure;
6 use Illuminate\Http\Request;
7 use Illuminate\Support\Facades\Auth;
8
9 class AdminAuthenticated
10 {
11     /**
12      * Handle an incoming request.
13      *
14      * @param \Illuminate\Http\Request $request
15      * @param \Closure(\Illuminate\Http\Request): (\Illuminate\Http\Response|\Illuminate\Http\RedirectResponse)
16      * @return \Illuminate\Http\Response|\Illuminate\Http\RedirectResponse
17      */
18     public function handle(Request $request, Closure $next)
19     {
20         if (Auth::guard('admin')->user()) {
21             return $next($request);
22         }
23         if ($request->ajax() || $request->wantsJson()) {
24             return response('Unauthorized.', 401);
25         } else {
26             return redirect(route('adminLogin'));
27         }
28     }
29 }

```

- Step 4: Add Middleware in Kernel.php file
- Move to the following path
- `app/Http/Kernel.php`
- Now add the code as follow

```

1 protected $middlewareGroups = [
2     'web' => [
3         \App\Http\Middleware\EncryptCookies::class,
4         \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
5         \Illuminate\Session\Middleware\StartSession::class,
6         // \Illuminate\Session\Middleware\AuthenticateSession::class,
7         \Illuminate\View\Middleware\ShareErrorsFromSession::class,
8         \App\Http\Middleware\VerifyCsrfToken::class,
9         \Illuminate\Routing\Middleware\SubstituteBindings::class,
10    ],
11
12    'admin' => [
13        \App\Http\Middleware\EncryptCookies::class,
14        \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
15        \Illuminate\Session\Middleware\StartSession::class,
16        // \Illuminate\Session\Middleware\AuthenticateSession::class,
17        \Illuminate\View\Middleware\ShareErrorsFromSession::class,
18        \App\Http\Middleware\VerifyCsrfToken::class,
19        \Illuminate\Routing\Middleware\SubstituteBindings::class,
20    ],
21
22 ];
23
24 /**
25  * The application's route middleware.
26  *
27  * These middleware may be assigned to groups or used individually.
28  *
29  * @var array<string, class-string>
30  */
31 protected $routeMiddleware = [
32     'auth' => \App\Http\Middleware\Authenticate::class,
33     'adminauth' => \App\Http\Middleware\AdminAuthenticate::class,
34     'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
35     'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
36     'can' => \Illuminate\Auth\Middleware\Authorize::class,
37     'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
38     'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
39     'signed' => \Illuminate\Routing\Middleware\ValidateSignature::class,
40     'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
41     'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
42 ];

```

- Step 5: Add Admin Login Routes
- Go to the below-mentioned path
- routes/web.php
- Now add the below code

```
1 Route::group(['prefix' => 'admin', 'namespace' => 'Admin'], function () {
2     Route::get('/login', [AdminAuthController::class, 'getLogin'])->name('adminLogin');
3     Route::post('/login', [AdminAuthController::class, 'postLogin'])->name('adminLoginPost')
4
5     Route::group(['middleware' => 'adminauth'], function () {
6         Route::get('/', function () {
7             return view('welcome');
8         }->name('adminDashboard'));
9
10    });
11 });
```

- Step 6: Create AdminAuth controller
- Run the following command
- php artisan make:controller Admin/AuthController
- Then navigate to the following path
- app/Http/Controllers/Admin/AdminAuth Controller.php
- And add the below code snippet

```

1 <?php
2 namespace App\Http\Controllers\Admin;
3
4
5 use App\Http\Controllers\Controller;
6 use Illuminate\Http\Request;
7 use Illuminate\Support\Facades\Session;
8
9 class AdminAuthController extends Controller
10 {
11     public function getLogin() {
12         return view('admin.auth.login');
13     }
14
15     public function postLogin(Request $request)
16     {
17         $this->validate($request, [
18             'email' => 'required|email',
19             'password' => 'required',
20         ]);
21
22         if(auth()->guard('admin')->attempt(['email' => $request->input('email'), 'password'
23             $user = auth()->guard('admin')->user();
24             if($user->is_admin == 1){
25                 return redirect()->route('adminDashboard')->with('success', 'You are Logged i
26             }
27         } else {
28             return back()->with('error', 'Whoops! invalid email and password. ');
29         }
30     }
31
32     public function adminLogout(Request $request)
33     {
34         auth()->guard('admin')->logout();
35         Session::flush();
36         Session::put('success', 'You are logout successfully');
37         return redirect(route('adminLogin'));
38     }
39 }

```

- Step 7: Create a Blade file
- Go to the below path
- `resources\views\admin\auth\login.blade.php`
- And add the code as follows

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

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Admin Login</title>
</head>
```



```

<body class="vertical-layout vertical-menu-modern" data-open="click" data-menu="vertical-menu-modern"
data-col="" data-framework="laravel">
<div class="auth-wrapper auth-basic px-2">
<div class="auth-inner my-2">
<!-- Login basic -->
@if(\Session::get('success'))
<div class="alert alert-success alert-dismissible fade show" role="alert">
<div class="alert-body">
{{ \Session::get('success') }}
</div>
<button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
</div>
@endif
{{ \Session::forget('success') }}
@if(\Session::get('error'))
<div class="alert alert-danger alert-dismissible fade show" role="alert">
<div class="alert-body">
{{ \Session::get('error') }}
</div>
<button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
</div>
@endif
<div class="card mb-0">
<div class="card-body">
<h2 class="brand-text text-primary ms-1">Admin Login</h2>

```

```

<form class="auth-login-form mt-2" action="{{route('adminLoginPost')}}" method="post">
    @csrf
    <div class="mb-1">
        <label for="email" class="form-label">Email</label>
        <input type="text" class="form-control" id="email" name="email" value="{{old('email')}}" autofocus />
        @if ($errors->has('email'))
            <span class="help-block font-red-mint">
                <strong>{{ $errors->first('email') }}</strong>
            </span>
        @endif
    </div>

    <div class="mb-1">
        <div class="d-flex justify-content-between">
            <label class="form-label" for="password">Password</label>
            <a href="{{url('auth/forgot-password-basic')}}">
                <small>Forgot Password?</small>
            </a>
        </div>
    </div>

```

```

<div class="input-group input-group-merge form-password-toggle">
  <input type="password" class="form-control form-control-merge" id="password"
name="password" tabindex="2" />
  <span class="input-group-text cursor-pointer"><i>eye"></i></span>
</div>
@if ($errors->has('password'))
  <span class="help-block font-red-mint">
    <strong>{{ $errors->first('password') }}</strong>
  </span>
@endif
</div>
<button type="submit" class="btn btn-primary w-100" tabindex="4">Sign in</button>
</form>
</div>
</div>
<!-- /Login basic -->
</div>
</div>
</body>
</html>

```

- Run your Laravel application
- Now, the admin auth is ready for the login. just run your Laravel application using the following command in the terminal.
- `php artisan serve`

# Thank You

