composer create-project --prefer-dist laravel/laravel projectmanagement

cd projectmanagement

composer require laravel/ui

php artisan ui bootstrap

php artisan ui bootstrap --auth  
 [npm install && npm run dev

npm install && npm run dev // again

npm run watch // if need

composer require laravelcollective/html

composer.phar require intervention/image

[

config/app.php under $providers

Intervention\Image\ImageServiceProvider::class,

config/app.php under $aliases

'Image' => Intervention\Image\Facades\Image::class,

]

php artisan vendor:publish --provider="Intervention\Image\ImageServiceProviderLaravelRecent"

$ php artisan make:model Role -mcr

$ php artisan make:model Permission -mcr

$ php artisan make:model Role\_has\_permission -m

$ php artisan make:controller AdminController

$ php artisan make:controller DeveloperController

$ php artisan make:controller ExecutiveController

**Inside config/app.php**

replace timezone

'timezone' => 'Asia/Dhaka',

**Inside app/Providers\** **AppServiceProvider.php**

use Illuminate\Pagination\Paginator;

  public function boot()

    {

         Paginator::useBootstrap(); // add this line for pagination

    }

**Edit all table**

User table

  Schema::create('users', function (Blueprint $table) {  
     $table->id();

            $table->string('name');

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

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

            $table->string('password');

            $table->tinyInteger('role\_id')->nullable();

            $table->tinyInteger('status\_id')->default('0');

            $table->string('profile\_image')->nullable();

            $table->rememberToken();

            $table->timestamps();

        });

Role table

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

            $table->id();

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

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

            $table->timestamps();

        });

Permission table

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

            $table->id();

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

            $table->timestamps();

        });

Role has permission table

Schema::create('role\_has\_permissions', function (Blueprint $table) {

            $table->bigIncrements('id');

            $table->unsignedBigInteger('role\_id')->nullable();

            $table->unsignedBigInteger('permission\_id')->nullable();

            $table->foreign('role\_id')->references('id')->on('roles')->onDelete('cascade');

            $table->foreign('permission\_id')->references('id')->on('permissions')->onDelete('cascade');

            $table->timestamps();

        });

**All model**   
===========

User model

 protected $fillable = [

        'name',

        'email',

        'password',

        'role\_id',

        'status\_id',

'profile\_image'

    ];

    public function roles(){

        return $this->belongsTo('App\Role');

    }

Role model

  protected $fillable=['name','slug'];

    public function users(){

            return $this->hasMany('App\User');

    }

Permission model

 protected $fillable=['name'];

    public function role(){

        return $this->hasMany('App\Role');

    }

Role has permission model

 protected $fillable=['role\_id','permission\_id'];

    public function roles()

        {

             return $this->belongsToMany(Role::class);

        }

    public function permissions()

        {

            return $this->hasMany('App\Permission');

        }

**Edit .env  
========**

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=projectmanagement

DB\_USERNAME=root

DB\_PASSWORD=

**Some command**  
==================

$ php artisan migrate

$ php artisan make:seed UsersTableSeeder

$ php artisan make:seed PermissionsTableSeeder

$ php artisan make:seed RolesTableSeeder

$ php artisan make:seed RoleHasPermissionsTableSeeder

**Create seeder**  
================

**Role seeder**  
---------------

use DB;

   public function run()

    {

           DB::table('roles')->insert([

            'name' => 'admin',

            'slug' => 'admin',

        ]);

        DB::table('roles')->insert([

            'name' => 'executive',

            'slug' => 'executive',

        ]);

        DB::table('roles')->insert([

            'name' => 'developer',

            'slug' => 'developer',

        ]);

    }

**Permission seeder**  
-------------------------

use App\Models\Permission;

use DB;

  $data = [

'menu-users',

            'menu-roles',

            'menu-permissions',

            'role-list',

            'role-create',

            'role-edit',

            'role-delete',

            'user-list',

            'user-create',

            'user-edit',

            'user-delete',

            'permission-list',

            'permission-create',

            'permission-edit',

            'permission-delete',

        ];

        foreach ($data as $permission) {

             Permission::create(['name' => $permission]);

        }

**User seeder**

use DB;

 DB::table('users')->insert([

            'role\_id' => '1',

            'status\_id' => '1',

            'name' => 'admin',

            'email' => 'admin@gmail.com',

            'password' => bcrypt('123456789'),

        ]);

        DB::table('users')->insert([

            'role\_id' => '2',

            'status\_id' => '1',

            'name' => 'executive',

            'email' => 'executive@gmail.com',

            'password' => bcrypt('123456789'),

        ]);

        DB::table('users')->insert([

            'role\_id' => '3',

            'status\_id' => '1',

            'name' => 'developer',

            'email' => 'developer@gmail.com',

            'password' => bcrypt('123456789'),

        ]);

**RoleHasPermmision Seeder**

use DB;

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '1',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '2',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '3',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '4',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '5',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '6',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '9',

]);

DB::table('role\_has\_permissions')->insert([

'role\_id' => '1',

'permission\_id' => '13',

]);

**Database seeder**

use DB;

        $this->call(PermissionsTableSeeder::class);

        $this->call(RolesTableSeeder::class);

        $this->call(UsersTableSeeder::class);

$this->call(RoleHasPermissionsTableSeeder::class);

$php artisan db:seed

Or

$ php artisan db:seed --class = UsersTableSeeder

$ php artisan db:seed --class = RolesTableSeeder

$ php artisan db:seed --class = PermissionsTableSeeder

$ php artisan db:seed --class = RoleHasPermissionsTableSeeder

**Create Middleware**   
=====================

$ php artisan make:middleware Admin

$ php artisan make:middleware Executive

$ php artisan make:middleware Developer

$ php artisan make:middleware PriventBackHistory

**app\Http\Middleware**  
------------------------------

Admin middleware

use Auth;

if(!Auth::check()){

            return redirect()->route('login');

        }

        // Role\_id=1 (admin)

        if(Auth::check() && Auth::user()->role\_id==1){

            return $next($request);

        }

        // Role\_id=2 (Executive)

        if(Auth::check() && Auth::user()->role\_id == 2){

            return redirect()->route('executive');

        }

        // Role\_id = 3 (Developer)

        if(Auth::check() && Auth::user()->role\_id == 3 ){

            return redirect()->route('developer');

        }

Executive Middleware

-----------------------------

use Auth;

  if(!Auth::check()){

            return redirect()->route('login');

        }

        // Role\_id=2 (Executive)

        if(Auth::check() && Auth::user()->role\_id == 2){

            return $next($request);

        }

Developer middleware

-----------------------------

use Auth;

      if(!Auth::check()){

            return redirect()->route('login');

        }

        // Role\_id = 3 (Developer)

        if(Auth::check() && Auth::user()->role\_id == 3 ){

            return $next($request);

        }

PriventBackHistory middleware

-------------------------------------------

use Auth;

     return $next($request);

        return $response->header('Cache-Control', 'private,no-cache,no-store,max-age=0,must-revalidate,pre-check=0,post-check=0')

                        ->header('Pragme','no-cashe')

                        ->header('Expires','Sun, 02 Jan 1990 00:00:00 GMT');

Open RedirectIfAuthenticated middleware (Just edit)   
---------------------------------------------------

  public function handle(Request $request, Closure $next, ...$guards)

    {

        if (Auth::guard($guards)->check() && Auth::user()->role\_id == 1) {

            return redirect()->route('admin.index');

        } elseif(Auth::guard($guards)->check() && Auth::user()->role\_id == 2){

            return redirect()->route('executive.index');

        } elseif(Auth::guard($guards)->check() && Auth::user()->role\_id == 3){

            return redirect()->route('developer.index');

        }else {

            return $next($request);

      }

        // $guards = empty($guards) ? [null] : $guards;

        // foreach ($guards as $guard) {

        //     if (Auth::guard($guard)->check()) {

        //         return redirect(RouteServiceProvider::HOME);

        //     }

        // }

        // return $next($request);

    }

**Kernel.php (Added last lines)**   
=================================

**app\Http**  
-------------

 protected $routeMiddleware = [

    'admin' => \App\Http\Middleware\Admin::class,

          'executive' => \App\Http\Middleware\Executive::class,

          'developer' => \App\Http\Middleware\Developer::class,

          'priventBackHistory' => \App\Http\Middleware\PriventBackHistory::class,

    ];

**LoginController**  
==================

**app\Http\Controllers\Auth\** **LoginController**

---------------------------------------

use Auth;

//at first comment below line

// protected $redirectTo = RouteServiceProvider::HOME;

// than past below code

    public function redirectTo(){

        switch(Auth::user()->role\_id){

            case 1:

                $this->redirectTo = '/admin';

                return $this->redirectTo;

            break;

            case 2:

                $this->redirectTo = '/executive';

                return $this->redirectTo;

            break;

            case 3:

                $this->redirectTo = '/developer';

                return $this->redirectTo;

            break;

            default;

                $this->redirectTo = '/login';

                return $this->redirectTo;

            break;

        }

    }

**RegisterController**  
=====================  
**app\Http\Controllers\Auth\** **RegisterController**--------------------------------------

// change ‘guest’ and write ‘auth’

  // public function \_\_construct()

    // {

    //     $this->middleware('guest');

    // }

public function \_\_construct()

    {

        $this->middleware('auth');

    }

*[ Copy and paste all controller(CustomAuth, User, Role, Permission, RoleHasPermission, Admin, Executive, Developer, so on….) from github ]*

**Route**  
=======

use App\Http\Controllers\CustomAuthController;

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

    return view('welcome');

});

// Auth::routes();

Auth::routes(['register' => false]);  // Can`t registrition with out login

Route::get('login', [CustomAuthController::class, 'index'])->name('login');

Route::post('custom-login', [CustomAuthController::class, 'customLogin'])->name('login.custom');

Route::get('registration', [CustomAuthController::class, 'registration'])->name('register-user');

Route::post('custom-registration', [CustomAuthController::class, 'customRegistration'])->name('register.custom');

Route::get('signout', [CustomAuthController::class, 'signOut'])->name('signout');

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

    Route::get('/', [App\Http\Controllers\AdminController::class, 'index'])->name('index');

    // Users

    Route::get('users', [App\Http\Controllers\AdminController::class, 'user'])->name('users');

    Route::get('/users.create',  [App\Http\Controllers\AdminController::class, 'usercreate'])->name('users.create');

    Route::post('users.store', [App\Http\Controllers\AdminController::class, 'userstore'])->name('users.store');

    Route::get('users.show.{id}', [App\Http\Controllers\AdminController::class, 'usershow'])->name('users.show');

    Route::get('users.edit.{id}', [App\Http\Controllers\AdminController::class, 'useredit'])->name('users.edit');

    Route::get('users.publish.{id}', [App\Http\Controllers\AdminController::class, 'userpublish'])->name('users.publish');

    Route::get('users.unpublish.{id}', [App\Http\Controllers\AdminController::class, 'userunpublish'])->name('users.unpublish');

    Route::patch('users.update.{id}', [App\Http\Controllers\AdminController::class, 'userupdate'])->name('users.update');

    Route::delete('/users.destroy.{id}', [App\Http\Controllers\AdminController::class, 'userdestroy'])->name('users.destroy');

    Route::post('/users.upload', [App\Http\Controllers\AdminController::class, 'usersupload'])->name('users.upload');

    Route::get('/users.fetch', [App\Http\Controllers\AdminController::class, 'usersfetch'])->name('users.fetch');

    Route::get('/users.delete', [App\Http\Controllers\AdminController::class, 'usersuploaddelete'])->name('users.delete');

    Route::post('/users.search', [App\Http\Controllers\AdminController::class, 'userssearch'])->name('users.search');

    // Admin role

    Route::get('/roles', [App\Http\Controllers\AdminController::class, 'role'])->name('roles');

    Route::get('/roles.create', [App\Http\Controllers\AdminController::class, 'rolecreate'])->name('roles.create');

    Route::post('/roles.store', [App\Http\Controllers\AdminController::class, 'rolestore'])->name('roles.store');

    Route::get('/roles.show.{id}', [App\Http\Controllers\AdminController::class, 'roleshow'])->name('roles.show');

    Route::get('/roles.edit.{id}', [App\Http\Controllers\AdminController::class, 'roleedit'])->name('roles.edit');

    Route::patch('/roles.update.{id}', [App\Http\Controllers\AdminController::class, 'roleupdate'])->name('roles.update');

    Route::delete('/roles.destroy.{id}', [App\Http\Controllers\AdminController::class, 'roledelete'])->name('roles.destroy');

    // Media

    Route::get('/media', [App\Http\Controllers\AdminController::class, 'media'])->name('media');

    Route::post('/media.upload', [App\Http\Controllers\AdminController::class, 'mediaupload'])->name('media.upload');

    Route::get('/media.fetch', [App\Http\Controllers\AdminController::class, 'mediafetch'])->name('media.fetch');

    Route::get('/media.delete', [App\Http\Controllers\AdminController::class, 'mediauploaddelete'])->name('media.delete');

    Route::post('/media.search', [App\Http\Controllers\AdminController::class, 'mediasearch'])->name('media.search');

    // Project

    Route::get('/projects', [App\Http\Controllers\AdminController::class, 'projects'])->name('projects');

    Route::get('/projects.create', [App\Http\Controllers\AdminController::class, 'projectscreate'])->name('projects.create');

    Route::post('/projects.store', [App\Http\Controllers\AdminController::class, 'projectsstore'])->name('projects.store');

    Route::post('/projects.upload', [App\Http\Controllers\AdminController::class, 'projectsupload'])->name('projects.upload');

    Route::get('/projects.fetch', [App\Http\Controllers\AdminController::class, 'projectsfetch'])->name('projects.fetch');

    Route::get('/projects.delete', [App\Http\Controllers\AdminController::class, 'uploaddelete'])->name('projects.delete');

    // Route::get('/permissions.show.{id}', [App\Http\Controllers\AdminController::class, 'permissionshow'])->name('permissions.show');

    Route::get('/projects.edit.{id}', [App\Http\Controllers\AdminController::class, 'projectsedit'])->name('projects.edit');

    Route::patch('/projects.update.{id}', [App\Http\Controllers\AdminController::class, 'projectsupdate'])->name('projects.update');

    Route::delete('/projects.destroy.{id}', [App\Http\Controllers\AdminController::class, 'projectsdelete'])->name('projects.destroy');

    Route::get('/projects.imagedestroy.{id}', [App\Http\Controllers\AdminController::class, 'projectsimagedelete'])->name('projects.imagedestroy');

    Route::post('/projects.search', [App\Http\Controllers\AdminController::class, 'projectssearch'])->name('projects.search');

    Route::post('/projects.imagesearch', [App\Http\Controllers\AdminController::class, 'projectsimagesearch'])->name('projects.imageSearch');

    // Documents

    Route::get('/documents', [App\Http\Controllers\AdminController::class, 'documents'])->name('documents');

    Route::get('/documents.create', [App\Http\Controllers\AdminController::class, 'documentscreate'])->name('documents.create');

    Route::post('/documents.store', [App\Http\Controllers\AdminController::class, 'documentsstore'])->name('documents.store');

    Route::post('/documents.upload', [App\Http\Controllers\AdminController::class, 'documentsupload'])->name('documents.upload');

    Route::get('/documents.fetch', [App\Http\Controllers\AdminController::class, 'documentsfetch'])->name('documents.fetch');

    Route::get('/documents.delete', [App\Http\Controllers\AdminController::class, 'documentsuploaddelete'])->name('documents.delete');

    Route::get('/documents.edit.{id}', [App\Http\Controllers\AdminController::class, 'documentsedit'])->name('documents.edit');

    Route::patch('/documents.update.{id}', [App\Http\Controllers\AdminController::class, 'documentsupdate'])->name('documents.update');

    Route::delete('/documents.destroy.{id}', [App\Http\Controllers\AdminController::class, 'documentsdelete'])->name('documents.destroy');

    Route::get('/documents.filedestroy.{id}', [App\Http\Controllers\AdminController::class, 'documentsfiledelete'])->name('documents.filedestroy');

    Route::post('/documents.search', [App\Http\Controllers\AdminController::class, 'documentssearch'])->name('documents.search');

    Route::post('/documents.filesearch', [App\Http\Controllers\AdminController::class, 'documentfilessearch'])->name('documents.filesearch');

    Route::delete('/documents.destroy.{id}', [App\Http\Controllers\AdminController::class, 'documentsdelete'])->name('documents.destroy'); // Documents

    //Task

    Route::get('/tasks', [App\Http\Controllers\AdminController::class, 'tasks'])->name('tasks');

    Route::get('/tasks.create', [App\Http\Controllers\AdminController::class, 'taskscreate'])->name('tasks.create');

    Route::post('/tasks.store', [App\Http\Controllers\AdminController::class, 'tasksstore'])->name('tasks.store');

    Route::get('/tasks.edit.{id}', [App\Http\Controllers\AdminController::class, 'tasksedit'])->name('tasks.edit');

    Route::get('/tasks.show.{id}', [App\Http\Controllers\AdminController::class, 'tasksshow'])->name('tasks.show');

    Route::patch('/tasks.update.{id}', [App\Http\Controllers\AdminController::class, 'tasksupdate'])->name('tasks.update');

    Route::get('/tasks.searchdestroy.{id}', [App\Http\Controllers\AdminController::class, 'searchdestroy'])->name('tasks.searchdestroy');

    Route::post('/tasks.search', [App\Http\Controllers\AdminController::class, 'taskssearch'])->name('tasks.search');

    Route::delete('/tasks.destroy.{id}', [App\Http\Controllers\AdminController::class, 'tasksdelete'])->name('tasks.destroy');

    Route::get('/tasts.profile.{id}', [App\Http\Controllers\AdminController::class, 'tasksprofile'])->name('tasts.profile');

    // Admin permission

    Route::get('/permissions', [App\Http\Controllers\AdminController::class, 'permission'])->name('permissions');

    Route::get('/permissions.create', [App\Http\Controllers\AdminController::class, 'permissioncreate'])->name('permissions.create');

    Route::post('/permissions.store', [App\Http\Controllers\AdminController::class, 'permissionstore'])->name('permissions.store');

    Route::get('/permissions.show.{id}', [App\Http\Controllers\AdminController::class, 'permissionshow'])->name('permissions.show');

    Route::get('/permissions.edit.{id}', [App\Http\Controllers\AdminController::class, 'permissionedit'])->name('permissions.edit');

    Route::patch('/permissions.update.{id}', [App\Http\Controllers\AdminController::class, 'permissionupdate'])->name('permissions.update');

    Route::delete('/permissions.destroy.{id}', [App\Http\Controllers\AdminController::class, 'permissiondelete'])->name('permissions.destroy');

    Route::post('/permissions.search', [App\Http\Controllers\AdminController::class, 'permissionsearch'])->name('permissons.search');

    Route::get('/permissions.permissiondelete.{id}', [App\Http\Controllers\AdminController::class, 'deletepermission'])->name('permissions.permissiondelete');

});

Route::group(['prefix' => 'executive', 'as'=>'executive.', 'middleware' => ['auth','executive','priventBackHistory']], function() {

    Route::get('/', [App\Http\Controllers\ExecutiveController::class, 'index'])->name('executive');

    // Users

    Route::get('users', [App\Http\Controllers\ExecutiveController::class, 'user'])->name('users');

    Route::get('/users.create',  [App\Http\Controllers\ExecutiveController::class, 'usercreate'])->name('users.create');

    Route::post('users.store', [App\Http\Controllers\ExecutiveController::class, 'userstore'])->name('users.store');

    Route::get('users.show.{id}', [App\Http\Controllers\ExecutiveController::class, 'usershow'])->name('users.show');

    Route::get('users.edit.{id}', [App\Http\Controllers\ExecutiveController::class, 'useredit'])->name('users.edit');

    Route::get('users.publish.{id}', [App\Http\Controllers\ExecutiveController::class, 'userpublish'])->name('users.publish');

    Route::get('users.unpublish.{id}', [App\Http\Controllers\ExecutiveController::class, 'userunpublish'])->name('users.unpublish');

    Route::patch('users.update.{id}', [App\Http\Controllers\ExecutiveController::class, 'userupdate'])->name('users.update');

    Route::delete('/users.destroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'userdestroy'])->name('users.destroy');

    // Admin role

    Route::get('/roles', [App\Http\Controllers\ExecutiveController::class, 'role'])->name('roles');

    Route::get('/roles.create', [App\Http\Controllers\ExecutiveController::class, 'rolecreate'])->name('roles.create');

    Route::post('/roles.store', [App\Http\Controllers\ExecutiveController::class, 'rolestore'])->name('roles.store');

    Route::get('/roles.show.{id}', [App\Http\Controllers\ExecutiveController::class, 'roleshow'])->name('roles.show');

    Route::get('/roles.edit.{id}', [App\Http\Controllers\ExecutiveController::class, 'roleedit'])->name('roles.edit');

    Route::patch('/roles.update.{id}', [App\Http\Controllers\ExecutiveController::class, 'roleupdate'])->name('roles.update');

    Route::delete('/roles.destroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'roledelete'])->name('roles.destroy');

   // Media

    Route::get('/media', [App\Http\Controllers\ExecutiveController::class, 'media'])->name('media');

    Route::post('/media.upload', [App\Http\Controllers\ExecutiveController::class, 'mediaupload'])->name('media.upload');

    Route::get('/media.fetch', [App\Http\Controllers\ExecutiveController::class, 'mediafetch'])->name('media.fetch');

    Route::get('/media.delete', [App\Http\Controllers\ExecutiveController::class, 'mediauploaddelete'])->name('media.delete');

    Route::post('/media.search', [App\Http\Controllers\ExecutiveController::class, 'mediasearch'])->name('media.search');

    // Project

    Route::get('/projects', [App\Http\Controllers\ExecutiveController::class, 'projects'])->name('projects');

    Route::get('/projects.create', [App\Http\Controllers\ExecutiveController::class, 'projectscreate'])->name('projects.create');

    Route::post('/projects.store', [App\Http\Controllers\ExecutiveController::class, 'projectsstore'])->name('projects.store');

    Route::post('/projects.upload', [App\Http\Controllers\ExecutiveController::class, 'projectsupload'])->name('projects.upload');

    Route::get('/projects.fetch', [App\Http\Controllers\ExecutiveController::class, 'projectsfetch'])->name('projects.fetch');

    Route::get('/projects.delete', [App\Http\Controllers\ExecutiveController::class, 'uploaddelete'])->name('projects.delete');

    // Route::get('/permissions.show.{id}', [App\Http\Controllers\ExecutiveController::class, 'permissionshow'])->name('permissions.show');

    Route::get('/projects.edit.{id}', [App\Http\Controllers\ExecutiveController::class, 'projectsedit'])->name('projects.edit');

    Route::patch('/projects.update.{id}', [App\Http\Controllers\ExecutiveController::class, 'projectsupdate'])->name('projects.update');

    Route::delete('/projects.destroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'projectsdelete'])->name('projects.destroy');

    Route::get('/projects.imagedestroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'projectsimagedelete'])->name('projects.imagedestroy');

    Route::post('/projects.search', [App\Http\Controllers\ExecutiveController::class, 'projectssearch'])->name('projects.search');

    Route::post('/projects.imagesearch', [App\Http\Controllers\ExecutiveController::class, 'projectsimagesearch'])->name('projects.imageSearch');

    // Documents

    Route::get('/documents', [App\Http\Controllers\ExecutiveController::class, 'documents'])->name('documents');

    Route::get('/documents.create', [App\Http\Controllers\ExecutiveController::class, 'documentscreate'])->name('documents.create');

    Route::post('/documents.store', [App\Http\Controllers\ExecutiveController::class, 'documentsstore'])->name('documents.store');

    Route::post('/documents.upload', [App\Http\Controllers\ExecutiveController::class, 'documentsupload'])->name('documents.upload');

    Route::get('/documents.fetch', [App\Http\Controllers\ExecutiveController::class, 'documentsfetch'])->name('documents.fetch');

    Route::get('/documents.delete', [App\Http\Controllers\ExecutiveController::class, 'documentsuploaddelete'])->name('documents.delete');

    // Route::get('/permissions.show.{id}', [App\Http\Controllers\ExecutiveController::class, 'permissionshow'])->name('permissions.show');

    Route::get('/documents.edit.{id}', [App\Http\Controllers\ExecutiveController::class, 'documentsedit'])->name('documents.edit');

    Route::patch('/documents.update.{id}', [App\Http\Controllers\ExecutiveController::class, 'documentsupdate'])->name('documents.update');

    Route::delete('/documents.destroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'documentsdelete'])->name('documents.destroy');

    Route::get('/documents.filedestroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'documentsfiledelete'])->name('documents.filedestroy');

    Route::post('/documents.search', [App\Http\Controllers\ExecutiveController::class, 'documentssearch'])->name('documents.search');

    Route::post('/documents.filesearch', [App\Http\Controllers\ExecutiveController::class, 'documentfilessearch'])->name('documents.filesearch');

    Route::delete('/documents.destroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'documentsdelete'])->name('documents.destroy');

    // Admin permission

    Route::get('/permissions', [App\Http\Controllers\ExecutiveController::class, 'permission'])->name('permissions');

    Route::get('/permissions.create', [App\Http\Controllers\ExecutiveController::class, 'permissioncreate'])->name('permissions.create');

    Route::post('/permissions.store', [App\Http\Controllers\ExecutiveController::class, 'permissionstore'])->name('permissions.store');

    Route::get('/permissions.show.{id}', [App\Http\Controllers\ExecutiveController::class, 'permissionshow'])->name('permissions.show');

    Route::get('/permissions.edit.{id}', [App\Http\Controllers\ExecutiveController::class, 'permissionedit'])->name('permissions.edit');

    Route::patch('/permissions.update.{id}', [App\Http\Controllers\ExecutiveController::class, 'permissionupdate'])->name('permissions.update');

    Route::delete('/permissions.destroy.{id}', [App\Http\Controllers\ExecutiveController::class, 'permissiondelete'])->name('permissions.destroy');

    Route::post('/permissions.search', [App\Http\Controllers\ExecutiveController::class, 'permissionsearch'])->name('permissons.search');

    Route::get('/permissions.permissiondelete.{id}', [App\Http\Controllers\ExecutiveController::class, 'deletepermission'])->name('permissions.permissiondelete');

});

Route::group(['prefix' => 'developer','middleware' => ['auth','developer','priventBackHistory']], function() {

    Route::get('/', [App\Http\Controllers\DeveloperController::class, 'index'])->name('developer');

    Route::get('users', [App\Http\Controllers\DeveloperController::class, 'user'])->name('developer.users');

    Route::get('/users.create',  [App\Http\Controllers\DeveloperController::class, 'usercreate'])->name('developer.users.create');

    Route::post('users.store', [App\Http\Controllers\DeveloperController::class, 'userstore'])->name('developer.users.store');

    Route::get('users.show.{id}', [App\Http\Controllers\DeveloperController::class, 'usershow'])->name('developer.users.show');

    Route::get('users.edit.{id}', [App\Http\Controllers\DeveloperController::class, 'useredit'])->name('developer.users.edit');

    Route::get('users.publish.{id}', [App\Http\Controllers\DeveloperController::class, 'userpublish'])->name('developer.users.publish');

    Route::get('users.unpublish.{id}', [App\Http\Controllers\DeveloperController::class, 'userunpublish'])->name('developer.users.unpublish');

    Route::patch('users.update.{id}', [App\Http\Controllers\DeveloperController::class, 'userupdate'])->name('developer.users.update');

    Route::delete('/users.destroy.{id}', [App\Http\Controllers\DeveloperController::class, 'userdestroy'])->name('developer.users.destroy');

    // developer role

    Route::get('/roles', [App\Http\Controllers\DeveloperController::class, 'role'])->name('developer.roles');

    Route::get('/roles.create', [App\Http\Controllers\DeveloperController::class, 'rolecreate'])->name('developer.roles.create');

    Route::post('/roles.store', [App\Http\Controllers\DeveloperController::class, 'rolestore'])->name('developer.roles.store');

    Route::get('/roles.show.{id}', [App\Http\Controllers\DeveloperController::class, 'roleshow'])->name('developer.roles.show');

    Route::get('/roles.edit.{id}', [App\Http\Controllers\DeveloperController::class, 'roleedit'])->name('developer.roles.edit');

    Route::patch('/roles.update.{id}', [App\Http\Controllers\DeveloperController::class, 'roleupdate'])->name('developer.roles.update');

    Route::delete('/roles.destroy.{id}', [App\Http\Controllers\DeveloperController::class, 'roledelete'])->name('developer.roles.destroy');

    // developer permission

    Route::get('/permissions', [App\Http\Controllers\DeveloperController::class, 'permission'])->name('developer.permissions');

    Route::get('/permissions.create', [App\Http\Controllers\DeveloperController::class, 'permissioncreate'])->name('developer.permissions.create');

    Route::post('/permissions.store', [App\Http\Controllers\DeveloperController::class, 'permissionstore'])->name('developer.permissions.store');

    Route::get('/permissions.show.{id}', [App\Http\Controllers\DeveloperController::class, 'permissionshow'])->name('developer.permissions.show');

    Route::get('/permissions.edit.{id}', [App\Http\Controllers\DeveloperController::class, 'permissionedit'])->name('developer.permissions.edit');

    Route::patch('/permissions.update.{id}', [App\Http\Controllers\DeveloperController::class, 'permissionupdate'])->name('developer.permissions.update');

    Route::delete('/permissions.destroy.{id}', [App\Http\Controllers\DeveloperController::class, 'permissiondelete'])->name('developer.permissions.destroy');

    // Media

    Route::get('/media', [App\Http\Controllers\DeveloperController::class, 'media'])->name('developer.media');

    Route::post('/media.upload', [App\Http\Controllers\DeveloperController::class, 'mediaupload'])->name('developer.media.upload');

    Route::get('/media.fetch', [App\Http\Controllers\DeveloperController::class, 'mediafetch'])->name('developer.media.fetch');

    Route::get('/media.delete', [App\Http\Controllers\DeveloperController::class, 'mediauploaddelete'])->name('developer.media.delete');

    Route::post('/media.search', [App\Http\Controllers\DeveloperController::class, 'mediasearch'])->name('developer.media.search');

    // Project

    Route::get('/projects', [App\Http\Controllers\DeveloperController::class, 'projects'])->name('developer.projects');

    Route::get('/projects.create', [App\Http\Controllers\DeveloperController::class, 'projectscreate'])->name('developer.projects.create');

    Route::post('/projects.store', [App\Http\Controllers\DeveloperController::class, 'projectsstore'])->name('developer.projects.store');

    Route::post('/projects.upload', [App\Http\Controllers\DeveloperController::class, 'projectsupload'])->name('developer.projects.upload');

    Route::get('/projects.fetch', [App\Http\Controllers\DeveloperController::class, 'projectsfetch'])->name('developer.projects.fetch');

    Route::get('/projects.delete', [App\Http\Controllers\DeveloperController::class, 'uploaddelete'])->name('developer.projects.delete');

    // Route::get('/permissions.show.{id}', [App\Http\Controllers\DeveloperController::class, 'permissionshow'])->name('developer.permissions.show');

    Route::get('/projects.edit.{id}', [App\Http\Controllers\DeveloperController::class, 'projectsedit'])->name('developer.projects.edit');

    Route::patch('/projects.update.{id}', [App\Http\Controllers\DeveloperController::class, 'projectsupdate'])->name('developer.projects.update');

    Route::delete('/projects.destroy.{id}', [App\Http\Controllers\DeveloperController::class, 'projectsdelete'])->name('developer.projects.destroy');

    Route::post('/projects.search', [App\Http\Controllers\DeveloperController::class, 'projectssearch'])->name('developer.projects.search');

     // Documents

    Route::get('/documents', [App\Http\Controllers\DeveloperController::class, 'documents'])->name('developer.documents');

    Route::get('/documents.create', [App\Http\Controllers\DeveloperController::class, 'documentscreate'])->name('developer.documents.create');

    Route::post('/documents.store', [App\Http\Controllers\DeveloperController::class, 'documentsstore'])->name('developer.documents.store');

    Route::post('/documents.upload', [App\Http\Controllers\DeveloperController::class, 'documentsupload'])->name('developer.documents.upload');

    Route::get('/documents.fetch', [App\Http\Controllers\DeveloperController::class, 'documentsfetch'])->name('developer.documents.fetch');

    Route::get('/documents.delete', [App\Http\Controllers\DeveloperController::class, 'documentsuploaddelete'])->name('developer.documents.delete');

    // Route::get('/permissions.show.{id}', [App\Http\Controllers\DeveloperController::class, 'permissionshow'])->name('developer.permissions.show');

    Route::get('/documents.edit.{id}', [App\Http\Controllers\DeveloperController::class, 'documentsedit'])->name('developer.documents.edit');

    Route::patch('/documents.update.{id}', [App\Http\Controllers\DeveloperController::class, 'documentsupdate'])->name('developer.documents.update');

    Route::delete('/documents.destroy.{id}', [App\Http\Controllers\DeveloperController::class, 'documentsdelete'])->name('developer.documents.destroy');

    Route::post('/documents.search', [App\Http\Controllers\DeveloperController::class, 'documentssearch'])->name('developer.documents.search');

     //Task

    Route::get('/tasks', [App\Http\Controllers\DeveloperController::class, 'tasks'])->name('developer.tasks');

    Route::get('/tasks.create', [App\Http\Controllers\DeveloperController::class, 'taskscreate'])->name('developer.tasks.create');

    Route::post('/tasks.store', [App\Http\Controllers\DeveloperController::class, 'tasksstore'])->name('developer.tasks.store');

    Route::get('/tasks.edit.{id}', [App\Http\Controllers\DeveloperController::class, 'tasksedit'])->name('developer.tasks.edit');

    Route::get('/tasks.show.{id}', [App\Http\Controllers\DeveloperController::class, 'tasksshow'])->name('developer.tasks.show');

    Route::patch('/tasks.update.{id}', [App\Http\Controllers\DeveloperController::class, 'tasksupdate'])->name('developer.tasks.update');

    Route::get('/tasks.searchdestroy.{id}', [App\Http\Controllers\DeveloperController::class, 'searchdestroy'])->name('developer.tasks.searchdestroy');

    Route::post('/tasks.search', [App\Http\Controllers\DeveloperController::class, 'taskssearch'])->name('developer.tasks.search');

    Route::delete('/tasks.destroy.{id}', [App\Http\Controllers\DeveloperController::class, 'tasksdelete'])->name('developer.tasks.destroy');

});

Route::group(['middleware' => ['auth']], function() {

    Route::resource('users', App\Http\Controllers\UserController::class);

    Route::resource('roles', App\Http\Controllers\RoleController::class);

    Route::resource('permissions', App\Http\Controllers\PermissionController::class);

});

Now login your project  
==================  
<http://localhost/projectmanagement/public/login>

If not login please copy and paste CustomAuthController, AdminController, DeveloperController, ExecutiveController, PermissionController, RoleController, RoleHasPermissionController, UserController and also copy and paste public, resources folder after that you cant login please do the below work.

$ php artisan make:model ImageUpload -m

$ php artisan make:model Project -m

$ php artisan make:model Document -m

$ php artisan make:model Task -m

$ php artisan make:model TaskMeta -m

ImageUpload Table

           $table->string('name')->nullable();  
           $table->string('title')->nullable();  
 $table->string('alt')->nullable();

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

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

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

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

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

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

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

Project table

            $table->string('project\_name')->nullable();

            $table->string('project\_logo')->nullable();

            $table->string('image\_id')->nullable();

Document table

       $table->bigIncrements('id');

            $table->string('document\_title')->nullable();

            $table->unsignedBigInteger('project\_id')->unsigned();

            $table->string('document\_image\_id')->nullable();

            $table->foreign('project\_id')->on('projects')->references('id')->onDelete('cascade');

Task table

    $table->bigIncrements('id');

            $table->string('task\_title')->nullable();

            $table->text('task\_description')->nullable();

            $table->string('assign\_by')->nullable();

            $table->string('task\_progress')->nullable();

            $table->string('task\_status')->nullable();

            $table->date("start\_date")->nullable()->default(null);

            $table->date("end\_date")->nullable()->default(null);

            $table->string('user\_id')->nullable();

            $table->unsignedBigInteger('project\_id')->unsigned();

            $table->foreign('project\_id')->on('projects')->references('id')->onDelete('cascade');

TaskMeta table

  $table->bigIncrements('id');

            $table->string('user\_id')->nullable();

            $table->string('role\_id')->nullable();

            $table->string('task\_id')->nullable();

**Related all model**  
  
ImageUpload Model

  protected $fillable=['name','title','alt','caption','description','slug','username','path','status','extention','created\_at','updated\_at'];

Project Model

   protected $table = 'projects';

   protected $fillable=['project\_name','project\_logo','image\_id'];

Document Model

  protected $fillable=['document\_title','document\_image\_id','project\_id'];

    public function project(){

        return $this->hasOne(Project::class);

    }

Task Model

    protected $fillable=['task\_title','task\_description','assign\_by','task\_progress','task\_status','start\_date','end\_date','user\_id','project\_id'];

    public function project(){

        return $this->hasOne(Project::class);

    }

TaskMeta Model

    protected $fillable=['user\_id','role\_id','task\_id'];

$php artisan migreate

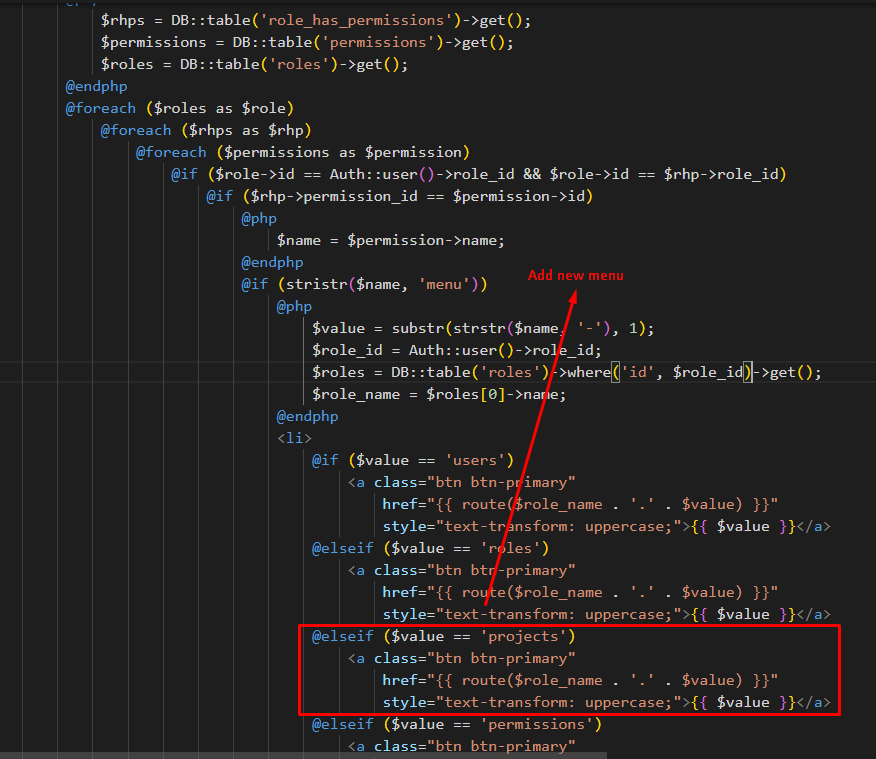
Now login and goto permission menu then add ‘menu-projects’, ‘menu-documents’,’ menu-tasks’,’ menu-media’ for left side menu here menu- add mandatory then go role menu click edit button then click menu and select required menu after that we can see menu. If we want to permission other menu (c r u d) for project, document and other. Then go permission menu and add (project-list, project-create, project-edit, project-delete, document-list, document-create, document-edit, document-delete, tasks-list, tasks-create, tasks-edit, tasks-delete, tasks-show, user-list, user-create, user-edit, user-delete, user-active and so on… here menu name mandatory like project, document, task, user so no… .Go to role edit menu and select project than click check button your needed.

Copy and paste recourse, public folder from github

**Custom menu add/edit**

**resources\views\admin\asset\nav.blade.php**

-----------------------------------------------------------------------------



<div class="col-md-3 left\_col">

         <div class="left\_col scroll-view">

             <div class="navbar nav\_title" style="border: 0;">

                 @php

                     $values = DB::table('users')

                         ->where('role\_id', Auth::user()->role\_id)

                         ->get();

                 @endphp

                 <a href="{{ asset($values[0]->name) }}"

                     class="site\_title text-center d-block"><span>Dashaboard</span></a>

             </div>

             <div class="clearfix"></div>

             <!-- sidebar menu -->

             <div id="sidebar-menu" class="main\_menu\_side hidden-print main\_menu">

                 <div class="menu\_section">

                     <ul class="nav side-menu">

                         @php

                             $rhps = DB::table('role\_has\_permissions')->get();

                             $permissions = DB::table('permissions')->get();

                             $roles = DB::table('roles')->get();

                         @endphp

                         @foreach ($roles as $role)

                             @foreach ($rhps as $rhp)

                                 @foreach ($permissions as $permission)

                                     @if ($role->id == Auth::user()->role\_id && $role->id == $rhp->role\_id)

                                         @if ($rhp->permission\_id == $permission->id)

                                             @php

                                                 $name = $permission->name;

                                             @endphp

                                             @if (stristr($name, 'menu'))

                                                 @php

                                                     $value = substr(strstr($name, '-'), 1);

                                                     $role\_id = Auth::user()->role\_id;

                                                     $nameRole = DB::table('roles')

                                                         ->where('id', $role\_id)

                                                         ->get();

                                                     $role\_name = $nameRole[0]->name;

                                                     //  print\_r($role\_name);

                                                 @endphp

                                                 <li>

                                                     @if ($value == 'users')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'roles')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'permissions')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'projects')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'media')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'documents')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'tasks')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @elseif ($value == 'dailyTask')

                                                         <a class="btn btn-primary"

                                                             href="{{ route($role\_name . '.' . $value) }}"

                                                             style="text-transform: uppercase;">{{ $value }}</a>

                                                     @else

                                                     @endif

                                                 </li>

                                             @endif

                                         @endif

                                     @endif

                                 @endforeach

                             @endforeach

                         @endforeach

                     </ul>

                 </div>

             </div>

             <!-- /sidebar menu -->

         </div>

     </div>

**resources\views\admin\asset\top.blade.php**

<!-- top navigation -->

  <div class="top\_nav">

      <div class="nav\_menu">

          <div class="nav toggle">

              <a id="menu\_toggle"><i class="fa fa-bars"></i></a>

          </div>

          <nav class="nav navbar-nav">

              <ul class=" navbar-right">

                  <li class="nav-item dropdown open" style="padding-left: 15px;">

                      <a href="javascript:;" class="user-profile dropdown-toggle" aria-haspopup="true"

                          id="navbarDropdown" data-toggle="dropdown" aria-expanded="false">

                          @php

                              $values = DB::table('users')

                                  ->where('id', Auth::user()->id)

                                  ->get();

                          @endphp

                          @if (!empty($values[0]->profile\_image))

                              <img src="{{ asset('thumbnail/' . $values[0]->profile\_image) }}" width="100px"

                                  height="80px" alt="" title="">

                          @else

                              <img src="{{ asset('images/profile/profile-image.png') }}" width="100px" height="80px"

                                  alt="" title="">

                          @endif

                          {{ $values[0]->name }}

                          {{-- <img src="{{ asset('/') }}images/img.jpg" alt="">{{ Auth::user()->id }} --}}

                      </a>

                      <div class="dropdown-menu dropdown-usermenu pull-right" aria-labelledby="navbarDropdown">

                          <a class="dropdown-item"

                              href="{{ route($values[0]->name . '.users.show', $values[0]->id) }}">

                              Profile</a>

                          <a class="dropdown-item"

                              href="{{ route($values[0]->name . '.users.edit', $values[0]->id) }}">

                              <span>Settings</span>

                          </a>

                          <a class="dropdown-item" href="{{ route('logout') }}"

                              onclick="event.preventDefault();

                               document.getElementById('logout-form').submit();"><i

                                  class="fa fa-sign-out pull-right"></i> Log

                              Out</a>

                          <form id="logout-form" action="{{ route('logout') }}" method="POST" class="d-none">

                              @csrf

                          </form>

                      </div>

                  </li>

              </ul>

          </nav>

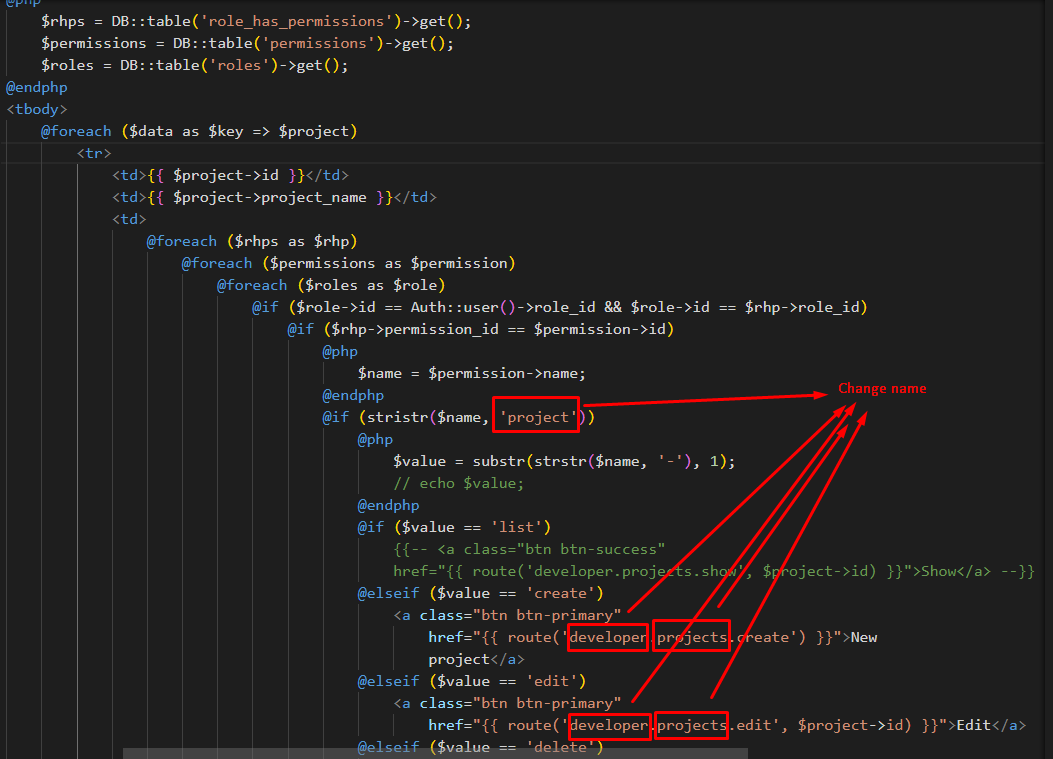
      </div>

  </div>

  <!-- /top navigation -->

**resources\views\developer\project\index.blade.php**

-----------------------------------------------------------------------------



Index.blade.php

 @extends('layouts.deshboard')

@section('content')

    <div class="container">

        <div class="justify-content-center">

            @if (\Session::has('success'))

                <div class="alert alert-success">

                    <p>{{ \Session::get('success') }}</p>

                </div>

            @endif

            @if (\Session::has('worning'))

                <div class="alert alert-danger">

                    <p>{{ \Session::get('worning') }}</p>

                </div>

            @endif

            <div class="card">

                <div class="card-header">Porject

                    @php

                        $role\_id = Auth::user()->role\_id;

                        $rhps = DB::table('role\_has\_permissions')

                            ->where('role\_id', $role\_id)

                            ->get();

                        $permissions = DB::table('permissions')->get();

                        foreach ($rhps as $rhpsvalue) {

                            $permissionId = $rhpsvalue->permission\_id;

                            foreach ($permissions as $pvalue) {

                                $pid = $pvalue->id;

                                if ($permissionId == $pid) {

                                    // print\_r($pvalue->name);

                                    if ($pvalue->name == 'project-create') {

                                        echo ' <span class="float-right">';

                                        echo '<a class="btn btn-primary btn-sm" href="projects.create"><i class="fas fa-plus"></i> New Project</a>';

                                        echo '</span>';

                                    }

                                }

                            }

                        }

                    @endphp

                    <span class="float-right">

                        <button id="search\_toggle" class="btn btn-info btn-sm"> <i class="fas fa-search"></i> Search

                        </button>

                    </span>

                </div>

                <div id="search\_third" style=" display: none;">

                    <div class="jumbotron justify-content-center">

                        <div class="from-group">

                            <input type="text" name="seach" class="form-control" id="input\_search\_image\_file"

                                placeholder="Search...">

                        </div>

                    </div>

                </div>

                <div class="card-body" id="index\_field">

                    <table class="table table-hover project-hidden">

                        <thead class="thead-dark">

                            <tr>

                                <th>#</th>

                                <th>Name</th>

                                <th>Logo</th>

                                <th>Create Date</th>

                                <th width="200px" class="text-center">Action</th>

                            </tr>

                        </thead>

                        @php

                            $rhps = DB::table('role\_has\_permissions')->get();

                            $permissions = DB::table('permissions')->get();

                            $roles = DB::table('roles')->get();

                        @endphp

                        <tbody>

                            <div id="project\_search\_response"></div>

                            @foreach ($data as $key => $project)

                                <tr>

                                    <td class="align-middle">{{ $project->id }}</td>

                                    <td class="align-middle">{{ $project->project\_name }}</td>

                                    <td class="align-middle">

                                        @if (!empty($project->project\_logo))

                                            <div id="post\_select\_image\_container" class="post-select-image-container">

                                                <img src="{{ asset('images/' . $project->project\_logo) }}"

                                                    id="selected\_image\_file" alt="" width="100px" height="80px">

                                            </div>

                                        @else

                                            <div id="post\_select\_image\_container" class="post-select-image-container">

                                                <img src="{{ asset('images/profile/cemera.jpg') }}" width="100px"

                                                    height="80px" alt="" title="">

                                            </div>

                                        @endif

                                    </td>

                                    <td class="align-middle">

                                        {{ date('d-m-Y', strtotime($project->created\_at)) }}</td>

                                    <td class="text-center align-middle">

                                        @foreach ($rhps as $rhp)

                                            @foreach ($permissions as $permission)

                                                @foreach ($roles as $role)

                                                    @if ($role->id == Auth::user()->role\_id && $role->id == $rhp->role\_id)

                                                        @if ($rhp->permission\_id == $permission->id)

                                                            @php

                                                                $name = $permission->name;

                                                            @endphp

                                                            @if (stristr($name, 'project'))

                                                                @php

                                                                    $value = substr(strstr($name, '-'), 1);

                                                                    // echo $value;

                                                                @endphp

                                                                @if ($value == 'list')

                                                                    {{-- <a class="btn btn-success"

                                                                    href="{{ route('admin.projects.show', $project->id) }}">Show</a> --}}

                                                                @elseif ($value == 'create')

                                                                @elseif ($value == 'edit')

                                                                    <a class="btn btn-primary btn-sm"

                                                                        href="{{ route('admin.projects.edit', $project->id) }}"><i

                                                                            class="fas fa-edit"></i></a>

                                                                @elseif ($value == 'delete')

                                                                 {!! Form::open(['method' => 'DELETE', 'route' => ['admin.projects.destroy', $project->id], 'style' => 'display:inline-table']) !!}

                                                                    {{ Form::button('<i class="fa fa-trash"></i>', ['type' => 'submit', 'class' => 'btn btn btn-danger btn-sm']) }}

                                                                    {!! Form::close() !!}

                                                                @else

                                                                @endif

                                                            @endif

                                                        @endif

                                                    @endif

                                                @endforeach

                                            @endforeach

                                        @endforeach

                                    </td>

                                </tr>

                            @endforeach

                        </tbody>

                    </table>

                    {{ $data->appends($\_GET)->links() }}

                </div>

            </div>

        </div>

    </div>

    <script type="text/javascript">

        //search image

        $(document).on('input', '#input\_search\_image\_file', function() {

            // use for previews table hidden

            document.getElementsByClassName('project-hidden')[0].style.visibility =

                'hidden';

            var search = $(this).val();

            var data = {

                "search": search

            };

            $.ajaxSetup({

                headers: {

                    'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')

                }

            });

            $.ajax({

                type: "POST",

                url: "{{ route('admin.projects.imageSearch') }}",

                data: data,

                success: function(response) {

                    document.getElementById("project\_search\_response").innerHTML = response

                }

            });

        });

        const searchtargetDiv = document.getElementById("search\_third");

        const searchbtn = document.getElementById("search\_toggle");

        searchbtn.onclick = function() {

            if (searchtargetDiv.style.display == "block") {

                searchtargetDiv.style.display = "none";

            } else {

                searchtargetDiv.style.display = "block";

            }

        };

    </script>

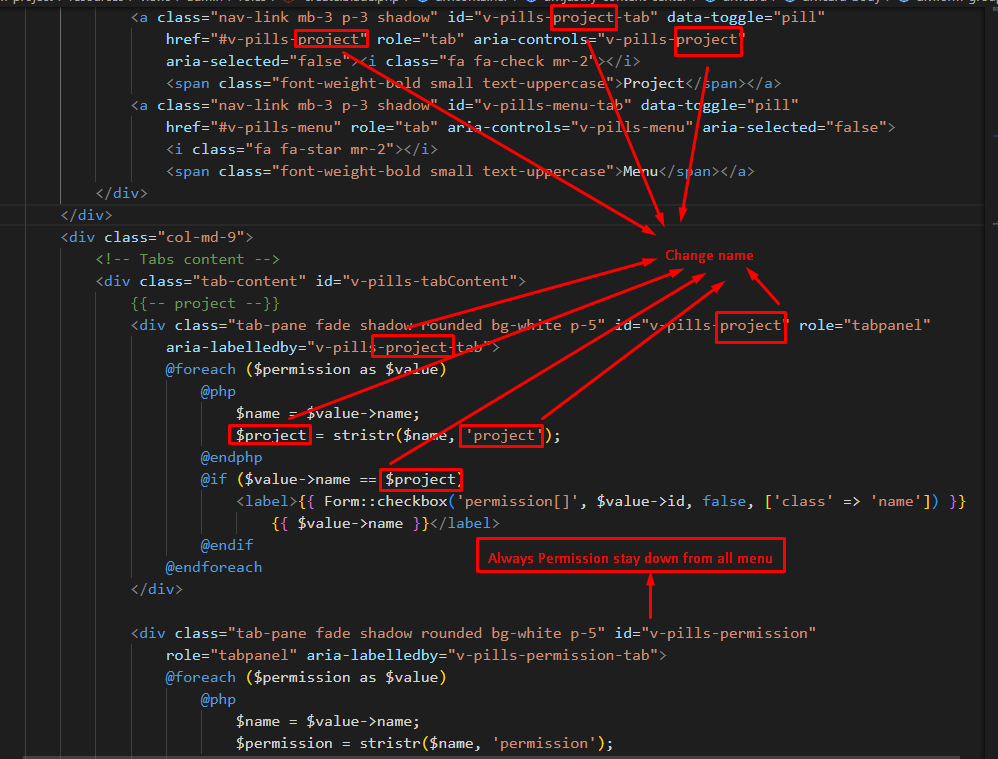
@endsection

Like above the way change all folder (project (C R U D), document (C R U D) and other)

**resources\views\admin\role\create.blade.php Or edit.blade.php**

------------------------------------------------------------------- -----------------------

create/edit blade file some change



    <div class="container">

        <div class="justify-content-center">

            @if (count($errors) > 0)

                <div class="alert alert-danger">

                    <strong>Opps!</strong> Something went wrong, please check below errors.<br><br>

                    <ul>

                        @foreach ($errors->all() as $error)

                            <li>{{ $error }}</li>

                        @endforeach

                    </ul>

                </div>

            @endif

            <div class="card">

                <div class="card-header">Create role

                    <span class="float-right">

                        <a class="btn btn-primary" href="{{ route('admin.roles') }}">Roles</a>

                    </span>

                </div>

                <div class="card-body">

                    {!! Form::open(['route' => 'admin.roles.store', 'method' => 'POST']) !!}

                    <div class="form-group">

                        <strong>User:</strong>

                        <select style="width:25%;margin-top: 10px;" name="role\_id" class="form-control">

                            @foreach ($users as $dt)

                                <option value="{{ $dt->id }}">{{ $dt->name }}</option>

                            @endforeach

                        </select>

                    </div>

                    <div class="form-group">

                        <strong class="mt-3"

                            style="text-align: center; font-size: 22px;">Permission:</strong>

                        <br />

                        <div class="col-md-3">

                            <!-- Tabs nav -->

                            <div class="nav flex-column nav-pills nav-pills-custom" id="v-pills-tab" role="tablist"

                                aria-orientation="vertical">

                                <a class="nav-link mb-3 p-3 shadow active" id="v-pills-user-tab" data-toggle="pill"

                                    href="#v-pills-user" role="tab" aria-controls="v-pills-user" aria-selected="true">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">User</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-role-tab" data-toggle="pill"

                                    href="#v-pills-role" role="tab" aria-controls="v-pills-role" aria-selected="false">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">User Role</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-project-tab" data-toggle="pill"

                                    href="#v-pills-project" role="tab" aria-controls="v-pills-project"

                                    aria-selected="false">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">Project</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-document-tab" data-toggle="pill"

                                    href="#v-pills-document" role="tab" aria-controls="v-pills-document"

                                    aria-selected="false">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">Document</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-media-tab" data-toggle="pill"

                                    href="#v-pills-media" role="tab" aria-controls="v-pills-media" aria-selected="false">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">Media</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-task-tab" data-toggle="pill"

                                    href="#v-pills-task" role="tab" aria-controls="v-pills-task" aria-selected="false">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">Task</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-permission-tab" data-toggle="pill"

                                    href="#v-pills-permission" role="tab" aria-controls="v-pills-permission"

                                    aria-selected="false">

                                    <i class="fa fa-check mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">Permission</span></a>

                                <a class="nav-link mb-3 p-3 shadow" id="v-pills-menu-tab" data-toggle="pill"

                                    href="#v-pills-menu" role="tab" aria-controls="v-pills-menu" aria-selected="false">

                                    <i class="fa fa-star mr-2"></i>

                                    <span class="font-weight-bold small text-uppercase">Menu</span></a>

                            </div>

                        </div>

                        <div class="col-md-9">

                            <!-- Tabs content -->

                            <div class="tab-content" id="v-pills-tabContent">

                                {{-- user --}}

                                <div class="tab-pane fade shadow rounded bg-white show active p-5" id="v-pills-user"

                                    role="tabpanel" aria-labelledby="v-pills-user-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $user = stristr($name, 'user');

                                        @endphp

                                        @if ($value->name == $user)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- role --}}

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-role" role="tabpanel"

                                    aria-labelledby="v-pills-role-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $role = stristr($name, 'role');

                                        @endphp

                                        @if ($value->name == $role)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- project --}}

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-project" role="tabpanel"

                                    aria-labelledby="v-pills-project-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $project = stristr($name, 'project');

                                        @endphp

                                        @if ($value->name == $project)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- document --}}

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-document" role="tabpanel"

                                    aria-labelledby="v-pills-document-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $document = stristr($name, 'document');

                                        @endphp

                                        @if ($value->name == $document)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- media --}}

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-media" role="tabpanel"

                                    aria-labelledby="v-pills-media-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $media = stristr($name, 'media');

                                        @endphp

                                        @if ($value->name == $media)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- task --}}

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-task" role="tabpanel"

                                    aria-labelledby="v-pills-task-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $task = stristr($name, 'task');

                                        @endphp

                                        @if ($value->name == $task)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- permission --}}

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-menu" role="tabpanel"

                                    aria-labelledby="v-pills-menu-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $menu = stristr($name, 'menu');

                                        @endphp

                                        @if ($value->name == $menu)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                <div class="tab-pane fade shadow rounded bg-white p-5" id="v-pills-permission"

                                    role="tabpanel" aria-labelledby="v-pills-permission-tab">

                                    @foreach ($permission as $value)

                                        @php

                                            $name = $value->name;

                                            $permission = stristr($name, 'permission');

                                        @endphp

                                        @if ($value->name == $permission)

                                            <label>{{ Form::checkbox('permission[]', $value->id, false, ['class' => 'name']) }}

                                                {{ $value->name }}</label>

                                        @endif

                                    @endforeach

                                </div>

                                {{-- Menu --}}

                            </div>

                            <button type="submit" class="btn btn-success col-md-6 float-right mr-3 mt-3">Submit</button>

                        </div>

                    </div>

                    {!! Form::close() !!}

                </div>

            </div>

        </div>

    </div>

Goto ROLES and click edit icon after that click menu and select required menu and click submit btn. After word again click ROLES and click edit icon then click required menu(project,document,task) and select require field then click submit.

Enjoy …