**Laravel Mega menu(CRUD)**

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

cd laravelmenu

composer require laravel/ui

php artisan ui bootstrap

php artisan ui bootstrap --auth

[npm install && npm run dev

npm run watch // if need

composer require laravelcollective/html

Php artisan make:model Category –mcr

Php artisan make:model Post –mcr

Php artisan make:model Postmeta –mcr // After delete controller

Php artisan make:model Page -mcr

Php artisan make:model Menu -mcr

Php artisan make:model Menuitem -mcr // After delete controller

**Category Table**

 $table->string('title')->nullable();

            $table->string('name');

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

            $table->integer('parent\_id')->nullable();

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

            $table->boolean('status')->default('1');

**Post Table**

    $table->bigIncrements('id');

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

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

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

            $table->text('excerpt')->nullable();

            $table->text('content')->nullable();

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

            $table->string('link')->unique()->nullable();

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

            $table->boolean('status')->default('1');

            $table->string('publish\_at')->nullable();

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

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

**Postmeta table**

         $table->bigIncrements('id');

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

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

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

**Page Table**

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

            $table->string('name');

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

            $table->string('link')->unique()->nullable();

            $table->integer('parent\_id')->nullable();

            $table->text('content')->nullable();

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

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

            $table->boolean('status')->default('1');

            $table->string('publish\_at')->nullable();

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

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

**Menu Table**

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

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

  $table->text('content')->nullable();

**Menu Item table**

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

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

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

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

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

            $table->integer('menu\_id')->nullable();

**Category Model**

protected $table = 'categories';

    public $timestamps = true;

    protected $fillable = ['title','name', 'slug', 'parent\_id','link','status'];

    public function subcategory()

        {

            return $this->hasMany(Category::class, 'parent\_id', 'id');

        }

    public function parent()

        {

            return $this->belongsTo(Category::class, 'parent\_id');

        }

**Post Model**

protected $table = 'posts';

    public $timestamps = true;

    protected $fillable = [

        'title',

        'slug',

        'excerpt',

        'content',

        'image',

        'link',

        'tag',

        'status',

        'publish\_at',

        'user\_id'];

    public function user()

        {

            return $this->belongsTo(User::class, 'id');

        }

**Postmeta Model**

  protected $table = 'postmetas';

    public $timestamps = true;

      protected $fillable = ['id','cat\_id','post\_id'];

    public function metareletion()

    {

          return $this->belongsTo(Category::class, 'id');

    }

**Page Model**

protected $table = 'pages';

    public $timestamps = true;

    protected $fillable = [

        'title',

        'name',

        'slug',

        'link',

        'parent\_id',

        'content',

        'image',

        'status',

        'publish\_at',

        'user\_id',

        'template',

    ];

    public function subpage()

        {

            return $this->hasMany(Page::class, 'parent\_id', 'id');

        }

    public function parent()

        {

            return $this->belongsTo(Page::class, 'parent\_id');

        }

    public function user()

        {

            return $this->belongsTo(User::class, 'id');

        }

**Menu Model**

 protected $fillable = ['title','location','content','created\_at','updated\_at'];

**Menuitem Model**

  protected $fillable = ['title','name','slug','type','target','menu\_id','created\_at','updated\_at'];

**Route (web.php)**

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

Route::get('/frontendindex',[App\Http\Controllers\frontController::class,'index']);

Route::get('/footer',[App\Http\Controllers\frontController::class,'footer']);

Route::get('/sidebar',[App\Http\Controllers\frontController::class,'sidebar']);

// Menu

Route::get('manage-menus/{id?}',[App\Http\Controllers\menuController::class,'index']);

Route::post('create-menu',[App\Http\Controllers\menuController::class,'store']);

Route::get('add-categories-to-menu',[App\Http\Controllers\menuController::class,'addCatToMenu']);

Route::get('add-page-to-menu',[App\Http\Controllers\menuController::class,'addPaseToMenu']);

Route::get('add-post-to-menu',[App\Http\Controllers\menuController::class,'addPostToMenu']);

Route::get('add-custom-link',[App\Http\Controllers\menuController::class,'addCustomLink']);

Route::get('save-menu',[App\Http\Controllers\menuController::class,'saveMenu']);

Route::get('update-menu',[App\Http\Controllers\menuController::class,'updateMenu']);

Route::post('update-menuitem/{id}',[App\Http\Controllers\menuController::class,'updateMenuItem']);

Route::get('delete-menuitem/{id}/{key}/{in?}',[App\Http\Controllers\menuController::class,'deleteMenuItem']);

Route::get('delete-menu/{id}',[App\Http\Controllers\menuController::class,'destroy']);

// category

Route::group(['prefix' => 'category'], function (){

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

        Route::get('/create', [App\Http\Controllers\CategoryController::class, 'create'])->name('category/create');

        Route::post('/store', [App\Http\Controllers\CategoryController::class, 'store'])->name('category/store');

        Route::get('/edit/{id}', [App\Http\Controllers\CategoryController::class, 'edit'])->name('category/edit');

        Route::post('/update', [App\Http\Controllers\CategoryController::class, 'update'])->name('category/update');

        Route::get('/delete/{id}', [App\Http\Controllers\CategoryController::class, 'destroy'])->name('category/delete');

        Route::get('/{category}', [App\Http\Controllers\CategoryController::class, 'categoryName'])->name('category');;

        Route::get('/publish/{id}', [App\Http\Controllers\CategoryController::class, 'publish'])->name('category/publish');

        Route::get('/unpublish/{id}', [App\Http\Controllers\CategoryController::class, 'unpublish'])->name('category/unpublish');

 });

 // post

Route::group(['prefix' => 'post'], function (){

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

        Route::get('/create', [App\Http\Controllers\PostController::class, 'create'])->name('post/create');

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

        Route::get('/edit/{id}', [App\Http\Controllers\PostController::class, 'edit'])->name('post/edit');

        Route::post('/update', [App\Http\Controllers\PostController::class, 'update'])->name('post/update');

        Route::get('/delete/{id}', [App\Http\Controllers\PostController::class, 'destroy'])->name('post/delete');

        Route::get('/publish/{id}', [App\Http\Controllers\PostController::class, 'publish'])->name('post/publish');

        Route::get('/unpublish/{id}', [App\Http\Controllers\PostController::class, 'unpublish'])->name('post/unpublish');

        Route::post('/multipledelete', [App\Http\Controllers\PostController::class, 'multipledelete'])->name('post/multipledelete');

        Route::get('/search', [App\Http\Controllers\PostController::class, 'search']);

        Route::get('/{post}', [App\Http\Controllers\PostController::class, 'postName'])->name('posts');;

 });

 // page

Route::group(['prefix' => 'page'], function (){

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

        Route::get('/create', [App\Http\Controllers\PageController::class, 'create'])->name('page/create');

        Route::post('/store', [App\Http\Controllers\PageController::class, 'store'])->name('page/store');

        Route::get('/edit/{id}', [App\Http\Controllers\PageController::class, 'edit'])->name('page/edit');

        Route::post('/update', [App\Http\Controllers\PageController::class, 'update'])->name('page/update');

        Route::get('/publish/{id}', [App\Http\Controllers\PageController::class, 'publish'])->name('page/publish');

        Route::get('/unpublish/{id}', [App\Http\Controllers\PageController::class, 'unpublish'])->name('page/unpublish');

        Route::get('/delete/{id}', [App\Http\Controllers\PageController::class, 'destroy'])->name('page/delete');;

        Route::post('/multipledelete', [App\Http\Controllers\PageController::class, 'multipledelete'])->name('page/multipledelete');

        Route::get('/search', [App\Http\Controllers\PageController::class, 'search']);

        Route::get('/{page}', [App\Http\Controllers\PageController::class, 'pagesName'])->name('pages');;

 });

Copy and paste all controller like (Category,Post,Page, Menu) code or replies these

htdocs\laravelmenu\app\Providers

**AppServiceProvider.php**

namespace App\Providers;

use Illuminate\Support\ServiceProvider;

use Illuminate\Pagination\Paginator;

use App\Models\menu;

use App\Models\menuitem;

use View;

use DB;

class AppServiceProvider extends ServiceProvider

{

    public function register()

    {

        //

    }

public function boot()

    {

        view()->composer(

            ['menu.header',

            'menu.footer',

             'menu.sidebar'],

                // '\*', // any where display

                function ($view) {

                $topNavItems='';

                $topNavItems2 ='';

                $topNavItems3 ='';

                $topNav = menu::where('location', 1)->first();

                $topNavItems = json\_decode($topNav->content);

                $topNavItems = $topNavItems[0];

                foreach($topNavItems as $menu){

                $menu->title = menuitem::where('id',$menu->id)->value('title');

                $menu->name = menuitem::where('id',$menu->id)->value('name');

                $menu->slug = menuitem::where('id',$menu->id)->value('slug');

                $menu->target = menuitem::where('id',$menu->id)->value('target');

                $menu->type = menuitem::where('id',$menu->id)->value('type');

                if(!empty($menu->children[0])){

                    foreach ($menu->children[0] as $child) {

                        $child->title = menuitem::where('id',$child->id)->value('title');

                        $child->name = menuitem::where('id',$child->id)->value('name');

                        $child->slug = menuitem::where('id',$child->id)->value('slug');

                        $child->target = menuitem::where('id',$child->id)->value('target');

                        $child->type = menuitem::where('id',$child->id)->value('type');

                             if(isset($child->children[0])){

                                foreach ($child->children[0] as $chil) {

                                    $chil->title = menuitem::where('id',$chil->id)->value('title');

                                    $chil->name = menuitem::where('id',$chil->id)->value('name');

                                    $chil->slug = menuitem::where('id',$chil->id)->value('slug');

                                    $chil->target = menuitem::where('id',$chil->id)->value('target');

                                    $chil->type = menuitem::where('id',$chil->id)->value('type');

                                            // print\_r($chil->title);

                                }

                            }

                        }

                    }

                }

                // ============Footer Menu=================

                 $topNav2 = menu::where('location',2)->first();

                 if(isset($topNav2->content)){

                $topNavItems2 = json\_decode($topNav2->content);

                $topNavItems2 = $topNavItems2[0];

                foreach($topNavItems2 as $menu){

                $menu->title = menuitem::where('id',$menu->id)->value('title');

                $menu->name = menuitem::where('id',$menu->id)->value('name');

                $menu->slug = menuitem::where('id',$menu->id)->value('slug');

                $menu->target = menuitem::where('id',$menu->id)->value('target');

                $menu->type = menuitem::where('id',$menu->id)->value('type');

                if(!empty($menu->children[0])){

                    foreach ($menu->children[0] as $child) {

                        $child->title = menuitem::where('id',$child->id)->value('title');

                        $child->name = menuitem::where('id',$child->id)->value('name');

                        $child->slug = menuitem::where('id',$child->id)->value('slug');

                        $child->target = menuitem::where('id',$child->id)->value('target');

                        $child->type = menuitem::where('id',$child->id)->value('type');

                            if(isset($child->children[0])){

                                foreach ($child->children[0] as $chil) {

                                    $chil->title = menuitem::where('id',$chil->id)->value('title');

                                    $chil->name = menuitem::where('id',$chil->id)->value('name');

                                    $chil->slug = menuitem::where('id',$chil->id)->value('slug');

                                    $chil->target = menuitem::where('id',$chil->id)->value('target');

                                    $chil->type = menuitem::where('id',$chil->id)->value('type');

                                            // print\_r($chil->title);

                                }

                            }

                        }

                    }

                }

             }

              // =============Sidebar================

                $topNav3 = menu::where('location', 3)->first();

            if($topNav3){

             $topNavItems3 = json\_decode($topNav3->content);

            $topNavItems3 = $topNavItems3[0];

            foreach($topNavItems3 as $menu){

              $menu->title = menuitem::where('id',$menu->id)->value('title');

              $menu->name = menuitem::where('id',$menu->id)->value('name');

              $menu->slug = menuitem::where('id',$menu->id)->value('slug');

              $menu->target = menuitem::where('id',$menu->id)->value('target');

              $menu->type = menuitem::where('id',$menu->id)->value('type');

              if(!empty($menu->children[0])){

                  foreach ($menu->children[0] as $child) {

                      $child->title = menuitem::where('id',$child->id)->value('title');

                      $child->name = menuitem::where('id',$child->id)->value('name');

                      $child->slug = menuitem::where('id',$child->id)->value('slug');

                      $child->target = menuitem::where('id',$child->id)->value('target');

                      $child->type = menuitem::where('id',$child->id)->value('type');

                       if(isset($child->children[0])){

                          foreach ($child->children[0] as $chil) {

                              $chil->title = menuitem::where('id',$chil->id)->value('title');

                              $chil->name = menuitem::where('id',$chil->id)->value('name');

                              $chil->slug = menuitem::where('id',$chil->id)->value('slug');

                              $chil->target = menuitem::where('id',$chil->id)->value('target');

                              $chil->type = menuitem::where('id',$chil->id)->value('type');

                                    // print\_r($chil->title);

                          }

                        }

                    }

                  }

                }

            }

                $view->with([

                    'topNavItems'=> $topNavItems,

                    'topNavItems2'=> $topNavItems2,

                    'topNavItems3'=> $topNavItems3]);

            }

        );

        Paginator::useBootstrap();

    }

}

htdocs\laravelmenu\resources\views

**welcome.blade.php**

# Copy and paste welcome.blade.php all code or replies it.

# Copy and paste all view file and folder.

# Copy and paste public folder (build, images, css, js, vendors).

|  |
| --- |
|  |
| First Register then Login Create menu name like(headermenu) |
|  |
|  |
| If you want add menu(category, page, post or custom link) in the menu structure |
|  |
| Select (headermenu,footermenu,sidebarmenu)->add menu->select location->move  Any menu his position->save menu |
| Move menu or submenu position here then click Save Menu |
|  |