convencion de anotacion: base nivel 1 y #numeracion y code nivel2 nivel3 error notas(~ significa renglon siguiente)

1º) crear las siete rutas necesarias para hacer nuestro crud

A-en admin.php creamos la ruta a) la configuramos en admilte.php

B-php artisan make:controller Admin/TagController -r

C- importamos ese controlador a nuestro archivo de rutas, luego le asignó el control de esa ruta a ese controlador, y como nombre Admin.tag

vamos a el controlador e (importamos el modelo TAG)

D-cambiamos el parametro id por (Tag $tag)

E-return c/u de los metodos index,create,show,edit (return view('admin.tag.index',xxx); xxx=compact('tag') para edit y show

F-creamos las vistas show,edit,create,index

\*\*\* nota OJO en vez de copiar-- un paso a paso pa segirlo y reproducir el resultado

G-EXTENDEMOS implamentamos la plantilla .:. la copiamos en c/u esta en resources\views\admin\index.blade.php

H) en adminlte.php configura rutas,iconos, (se borro otros links(multi level)

\*\*\* nota en index.blade.php deberiamos mostrar los post del usuario que esta autenticado y porveer una maner da busqueda (con un plugin da jsquery datable

I- nota trabajaremos con livewire para lo que en adminlte.php al final 'livewire' => false, a true

a-creamos un new componente php artisan make:livewire Admin/PostsIndex recordemos que este rendrizara a livewire.admin.posts-index'

b- pa poder utilizarlo en admin.blade.php en @section('content') @livewire('admin.posts-index')

c. en PostsIndex .php llamamos a use App\Models\Post; vamos a toma a PostsIndex.php como si fuera un clontrolador y

y recuperamos la INFO ->

seria 2º) $posts = Post::where('user\_id', auth()->user()->id)->paginate(); //pedimos que filtre de acuerdo al campo user-id

2º)-recuperamos INFO… en TagController.php en index( ) --$tag = Tag::all( ); // recupera datos -en

3º) pero en posts-index'.blade.php -> no hay que borrar… y creamos los div.card …etc según 3º y va con la mismo code … div.card~div.card-body… etc

Nota\*\*\*como el uesr de id=1 no tenia post no le aparecio nada - ->cambiamos en seeders pa que a menos user le toque mas posts y php artisan migrate:fresh –seed

b-queeremos q la paginacion aparesca al final -> debemos import (use Livewire\WithPagination;) y adentro use WithPagination; y en la vista debajo del card-body div.card-footer -- error pagina feo xq qfalta protected $paginationTheme = "bootstrap"; r 12

c-queremo un input search arriba pa filtrar la info -> en posts-index'.blade.php r4 un div.card-header~input.(form-control, placeholder)

d-queremos a medida que se escribe se filtre - ahi r13 public $search; -- y pedimos que se sincronice con esta c) cuando vamos al ultimo reg y lo metemos en search no aparece -> debemos add public function updatingSearch(){$this->resetPage();}--se activa solo cuando la propiedad search cambie

e- cuando no alla registros -> letrero quelo diga -> con un if encerramos toda la tabla

error mostraba el input pegado xq mal card estaba crad

3ª) queremo que los ultimos post aparesca de 1ero

3º)-en index.blade.php borramos las 2 ultimas seccione de las 5 y en @content --.card ~ table class table ~ table-striped ~ thead.tr.th\*3

(1ID,2A,NAME,3colspoan2) y tbody.@foreach.($tags as $tag).tr^td{{$tag->id}}-td{{$tag->name}}-td.<a class="btn btn-primary btn-sm"

href="{{route('a.t.edit',$tag)}}">editar<>-form action="{{route('a.t.s',$tag)}}" method="POST">-@csrf-@method('delete')-buttom

class="btn btn-danger btn=sm" type="submit">Eliminar< \*\*\*nota pa que amvos este a la derecha en td widtn="10px"

\*\*\*nota error1 undefine $tag cambio a tags pero igual RTO error1a falta ,compact('tags')) error2 boton editar no aparece RTO faltan class="btn..." error3 no color po fila rta a)tenia car es card-body b) el @csrf y @method(..) no estaban dentro del form c) tenia stripe y es striped

A- sidebar se mueva en adminlte.php r 88 'layout\_fixed\_sidebar' => null, a true (pa que no se mueve sidebar con contenido)

B-boton crear colocamos un enlace con forma de boton en la cabeza pa q nos redireccione a admon.tag.crear etiquetas

en index.blade.php @section('content-header') <a.class="btn...float-right" href="{{r('a.t.c',)}}>nueva etiqueta<a/>"

\*\*\*nota en category este boton se puso en @section('content') sin un card-header

4º)-en create.blade.php en @section('conten) card~card-body~ (L-collec o (L-c)) form::open(['route'=>'a.t.store'])~div class=form-group~form::leabel('name','Nombre')~ ~form::text('name',null,'class'=>'form-control','placeholder'=>'ingerese el nombre de la etiqueta...')~form::colse( )

\*\*\*L-collective a) el mande el @csrf b)si no le decimos el method x defauld es POST

B-SLUG duplicamos el div interior pa el slug .:. queda el div~div~form::close ...cambiamos 3 slug por tag y al final q diga slug

a-pegamos el plugin conversor de slug

D-en el input slug (,'readonly')evitar que el usuario pueda cambiar el slug

E-mostrar un select desplegable que permita al user escoger un color -> etiquetas (selct y option)

div.class="form-group"~label.for>color:<>~select.name="color" id=" class="form-control>~option\*3>a,b,c<

F-boton enviar -> a) form::submit. 'crear etiqueta,['class'=>'...} b) en TagController.php store( ) return $request->all( );

le pedimos q nos retur lo que se esta mandando por el formulario

G-select con collective form::label.'color','color:'~form::select.('color',$colors,null,['class'=>'form-control'] yyyy en TagController.php en create( ) $colors=['red'=>'color rojo',....] y ,compact('color'));

Cambia xq escogemos es categories PostController.php -use Categories- en create() { $cate=Categories::pluck('name');

en create copiar el select de tags

H- textArea pa colocar el extracto del post div.form-group~form::label('extract','extracto)~form::texarea('extract',null,'class=>form-control)~~~se repi lo anterior y pa body ~ form::submit('crear post',['class'=>'primary] \*\*\*nota la idea es permitir escribir texto enriquecido ahi ->Plugin(CKEditor)->bajamos el cdn e indicamos a cual tex area se aplica

i-@19:49 status opcion agregar (borrador, publicado) input radio div.form-group~<p>~label~from::radio

V- un boton al clic redirect a blog.text/admin/post/create y alli muestre un form -> en index.blade.php r5 @section('content\_header') add <a.btn.flow-right href="{{route(admin.posts.create en create.blade.php div.card~div.card-body~ un desplegable pa escoger cartegoria en PostController.php -use Categories- en create() { $cate=Categories::pluck('name'); en create copiar el select de tags 14:32 textArea pa colocar el extracto del post div.form-group~form::label('extract','extracto)~form::texarea('extract',null,'class=>form-control)~~~se repi lo anterior y

pa body ~ form::submit('crear post',['class'=>'primary] \*\*\*nota la idea es permitir escribir texto enriquecido ahi ->Plugin(CKEditor)->bajamos el cdn e indicamos a cual tex area se aplica

A-eEstado @19:49 status opcion agregar (borrador, publicado) input radio div.form-group~<p>~label~from::radio

si el usuario envia el form y ha marcado BORRADOR (que se le permita estar vacia(s)etiquetas, extracto, body si PUBLICADO -> todo lleno .:. dos validaciones

a) db null a extrac ^ body ^ etiquetas -> en database\migrations\2021\_04\_13\_232155\_create\_posts\_table.php b) las reglas las podriamos incluir en el controlador pero como sin un poco complejas xq depende de status usaremos un formRequest

c) creamos php artisan make:request StorePostRequest limitar a user el ingreso a cierta seccion si cumple cierta condicion ej: admini bloger BUTBUT esto lo manejaremos son L jpermmission -=> true ahi y en la RULES()

d) en postcontroller.php use el form request d1) en create.blade.php imprimir los mensajes de validacion @error ... --probamos y bien las 2 valida

e) dentro del form debemos integrar el id del user actual (pa relacionarlos) con un imput oculto hidde --un problema el usuario podria reemplazar el valor en el inspector pa impedirlo vovlemos a; formreques SotorePo.... si utilizaremos el authoriza()

f) guarda la info en db g) habilitar asignacion masiva dos maneras co fillable [los que queremos] y con guarder(colocamos los campos que queremos evitar que se llenen por asignaicon masiba)

\*\*\*nota cuando guardamos una parte se guardara en la tabla post pero otra va en la tabla relacion post\_tag

-> en PostController.php y ponemos una condicional si enviamos infor de etiquetas una vez que recupero la relacion ->attach([])

5º) en store( ) @27:26 en store( ) { $request->validation(['name=>'required',y a slug|unique:tags, y a color ]), // reglas de validacion

6º) en edit.blade.php //debajo de c/u input los posibles errores de validacion. mostramos en

\*\*\*nota pregunto si existe algún error de validación en el campo name en el caso de que lo hubiera vamos a poner dentro de una etiqueta small el mensaje -@error('name')<small class="text-danger>{{$message}}<

a-igual en slug

7º)- TagController.php en store( ) si pasa la regla de validacion -> me cree un new etiqueta $tag = Tag::create($request->all( )); //le pasamos la informacion q estamos enviando desde el formulario con ($request->all( ))

luego return redirect( )->route('a.ts.e',c)//;le pasamos como parametro ,compact('tag');

8º) en TagModels.php -protected fillable = ['name','slug','color'] //habilitar asignacion masiba en TagModels.php

error1 validation no validate si error2 tenia redirect('a.ts....) era redirect( )->route('a.ts....

9º)-en edit.blade.php div.card~div.card-body~form::model.$tag.'route'=>'route.tags.update' //1er paramt variable del registro q queremos actualizar . 2o pamtro la ruta y need a pamtro .:. ['ruta',pamtro $tag] y es con .:.['route'=>['a.t.u',$t],'method '=>'put']~form::close

\*\*\*nota y adentro todos los campos BUT BUT fastidoso =>

10º)- form.blade.php

A) 1º) creamos carpeta y vista @33:03 partials creamos carpeta y vista resources\views\admin\tags\partials\form.blade.php 2o) cortamos el form en r 29 renglone de create y copy en new 3o) incluimos el contenido de form.blade.php en create.blade.php mediante @include('name') //debemos pasarle la ruta relativa a la carpeta view o sea @include('admin.tags.partials.form') y too copy en edit.blade.php

B) nos dara un ERROR xq la $color solo esta en creat( ) -> 1) la copy y la pegarla en edit( ) y pasarsela ala vista ojo error xq no cuide que ya estaba pasando 'tag' y lo borre

11º) -en edit.blade.php form::submit.class'=>'btn btn-primay//debajo de los input ponemos un boton y al dar clic la info se ira a admin.tags.update

12º) regla de validation -en TagController.php update( ) A) ponemos regla de validation, lo copiamos como esta en store( ) yyyy B) sabemos que slug da problema => le traemos el arreglo add ,slug,$tag->id" ojo " al ini C) si pasa la regla de validacion, $tag->update($request->all( )) //le paso la inf que se esta enviando desde el form $request->all( ); D) yy return redirect( )->route('a.t.e',$tag)-> //le pasamos la info de la etiqueta $tag y va con un mensaje de session

13º) Éxito Mensaje ->with('info','exito en etique edit') Ea) copiamos el info desde edit de category

A) slug pugin y copiamos el puglin del slug borrando @section('js')

B) mostrar el mensaaje too al crear la etiqueta a) pegar en tag/create.blade.php debajo de @conten b)

14º) destroy( ) A) $tag->delete( ); B) return redirect( )->route('admin.tags.index')->with('info','la etiqueta se borro bien'); C) en index.blade.php @if (session('info')).... error 1 en eliminar estaba store y debe ser destroy

15º)-que se envie el slug y no el id en tagMpdels.php

16º)-pinte de azul el boton de etiquetas no solo caundo esta en index sino too en edit en adminlte.php r252 add 'active' => ['admin/tags\*'] a 'route' => 'admin.tags.index',