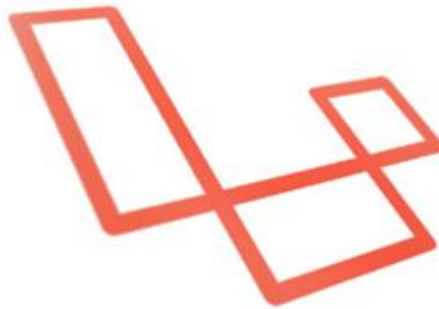


Desarrollo de Aplicaciones Empresariales

LABORATORIO N° 8

Laravel – Funcionalidad del Proyecto II

CODIGO DEL CURSO:



laravel

Alumno(s)		Nota
Grupo		
Ciclo		
Fecha de entrega		

I.- OBJETIVOS:

- Crear funcionalidad en la aplicación del Gestor de Imágenes.

II.- SEGURIDAD:**Advertencia:**

En este laboratorio está prohibida la manipulación del hardware, conexiones eléctricas o de red; así como la ingestión de alimentos o bebidas.

III.- FUNDAMENTO TEÓRICO:

Revise sus diapositivas del tema antes del desarrollo del laboratorio.

IV.- NORMAS EMPLEADAS:

No aplica

V.- RECURSOS:

- En este laboratorio cada alumno trabajará con un equipo con Windows 8.

VI.- METODOLOGÍA PARA EL DESARROLLO DE LA TAREA:

- El desarrollo del laboratorio es individual.

VII.- PROCEDIMIENTO:**Nota:**

Las secciones en cursivas son demostrativas, pero sirven para que usted pueda instalar las herramientas de desarrollo en un equipo externo.

CREANDO LA FUNCIONALIDAD PARA EDITAR UNA FOTO – MOVER Y COPIAR ARCHIVOS

1. Diríjase al controlador de las fotos y ubíquese en el action `getActualizarFoto()`.
2. Realice los siguientes campos en la función:

```
53     public function getActualizarFoto($id)
54     {
55         $foto=Foto::find($id);
56         return view('fotos.actualizar-foto',['foto'=>$foto]);
57     }
```

Explique: ¿Qué se hizo en esta función?

3. Cree una vista para Foto de nombre “**actualizar-foto.blade.php**”. Puede ayudarse copiando en código de **mostrar**. Modifique dicha vista para que su código quede de la siguiente manera:

```

1  @extends('app')
2
3  @section('content')
4  <div class="container-fluid">
5      <form class="form-horizontal" role="form" method="POST" action="/validado/fotos/actualizar-foto" enctype="multipart/form-data">
6          <input type="hidden" name="token" value="{{ csrf_token() }}" required>
7          <input type="hidden" name="id" value="{{ $foto->id }}" required>
8          <div class="form-group required">
9              <label class="col-md-4 control-label">Nombre</label>
10             <div class="col-md-6">
11                 <input type="text" class="form-control" name="nombre" value="{{ $foto->nombre }}" required>
12             </div>
13         </div>
14
15         <div class="form-group required">
16             <label class="col-md-4 control-label">Descripción</label>
17             <div class="col-md-6">
18                 <textarea type="text" class="form-control" name="descripcion" rows="3" required> {{ $foto->descripcion }} </textarea>
19             </div>
20         </div>
21
22         <div class="form-group required">
23             <label class="col-md-4 control-label">Imagen max: 20Mb</label>
24             <div class="col-md-6">
25                 <input type="file" class="form-control" name="imagen">
26             </div>
27         </div>
28         <div class="form-group">
29             <div class="col-md-6 col-md-offset-4">
30                 <button type="submit" class="btn btn-primary">
31                     Actualizar Imagen
32                 </button>
33             </div>
34         </div>
35     </form>
36 </div>
37 @endsection
38

```

Destaca en este formulario que ahora, el ID de la foto que enviamos se hace a través de un input oculto, ya no por url. Además, que el campo de la imagen, no es requerido, pues la persona podría o no actualizar la foto necesariamente.

- Diríjase a la vista de mostrar fotos, para agregar el botón de editar foto, hágalo de la siguiente manera:

```

1  @extends('app')
2
3  @section('content')
4  <div class="container-fluid">
5      <p><a href="/validado/fotos/crear-foto?id={{ $id }}" class="btn btn-primary" role="button">Crear Foto</a></p>
6      @if(sizeof($fotos)>0)
7          @foreach($fotos as $foto)
8              <div class="row">
9                  <div class="col-sm-6 col-md-12">
10                     <div class="thumbnail">
11                         
12                         <div class="caption">
13                             <h3>{{ $foto->nombre }}</h3>
14                             <p>{{ $foto->descripcion }}</p>
15                         </div>
16                         <p><a href="/validado/fotos/actualizar-foto/{{ $foto->id }}" class="btn btn-primary" role="button">Editar Foto</a></p>
17                     </div>
18                 </div>
19             </div>
20             @endforeach
21         @else
22             <div class="alert alert-danger">
23                 <p>Al parecer este álbum no tiene Fotos. Crea una!</p>
24             </div>
25         @endif
26     </div>
27 @endsection
28

```

Ya tenemos la parte funcional de conseguir la página para editar la foto. Ahora falta la acción de enviarla y guardarla

- Cree un nuevo Request de nombre “**ActualizarFotoRequest**”.
- No se olvide de referenciar a este Request en el controlador de **Foto**, que es donde se utilizará esta clase.
- Diríjase al Request creado recientemente.
- No se olvide de usar la definición de la librería **Auth**, en la cabecera del documento.
- En el Request edite el código para que quede de la siguiente manera:

```

1  <?php namespace GestorImagenes\Http\Requests;
2  use GestorImagenes\Http\Requests\Request;
3  use Illuminate\Support\Facades\Auth;
4  use GestorImagenes\Foto;
5  class ActualizarFotoRequest extends Request {
6      /**
7       * Determine if the user is authorized to make this request.
8       *
9       * @return bool
10      */
11     public function authorize()
12     {
13         $user=Auth::user();
14         $id=$this->get('id');
15         $foto=Foto::find($id);
16         $album=$user->albumes()->find($foto->album_id);
17
18         if($album){
19             return true;
20         }
21         return false;
22     }
23     /**
24      * Get the validation rules that apply to the request.
25      *
26      * @return array
27      */
28     public function rules()
29     {
30         return [
31             'id'=>'required|exists:fotos,id',
32             'nombre'=>'required',
33             'descripcion'=>'required',
34             'imagen'=>'image|max:20000'
35         ];
36     }
37 }
38

```

Explique: ¿Qué estamos haciendo en las dos funciones principales de este Request?

10. Ahora modifique la función postActualizarFoto para que quede de la siguiente manera.

```

59     public function postActualizarFoto(ActualizarFotoRequest $request){
60         $foto=Foto::find($request->get($id));
61         $foto->nombre=$request->get('nombre');
62         $foto->descripcion=$request->get('descripcion');
63         if($request->hasFile('imagen')){
64             $imagen=$request->file('imagen');
65             $ruta='/img/';
66             $nombre=sha1(Carbon::now()).'.'.$imagen->guessExtension();
67             $imagen->move(getcwd().$ruta,$nombre);
68             $rutaanterior=getcwd().$foto->ruta;
69             if(file_exists($rutaanterior))
70             {
71                 unlink(realpath($rutaanterior));
72             }
73             $foto->ruta=$ruta.$nombre;
74         }
75         $foto->save();
76         return redirect("/validado/fotos?id=$foto->album_id")->with('editada','La foto fue editada');
77     }
78

```

Explique Línea a línea lo que el código pretende hacer:

Ahora explique:

Desde la línea 69 (el bloque if), ¿por qué tiene que verificarse que la ruta exista?. Es necesario la verificación

11. Modifique la vista de Mostrar Fotos de modo que quede de la siguiente manera:

```
1 @extends('app')
2
3 @section('content')
4
5     @if(Session::has('editada'))
6         <div class="alert alert-success">
7             <p>La foto fué editada</p>
8         </div>
9     @endif
10
11     <div class="container-fluid">
12         <p><a href="/validado/fotos/crear-foto?id={{$id}}" class="btn btn-prim<
13     @if(sizeof($fotos)>0)
14         @foreach($fotos as $foto)
15             <div class="row">
16                 <div class="col-sm-6 col-md-12">
17                     <div class="thumbnail">
18                         
19                         <div class="caption">
20                             <p>{{ $foto->nombre }}</p>
```

¿Qué hace el código de la vista?

Pruebe el código para verificar que las fotos pueden ser editadas.

