

Desarrollo de Aplicaciones Empresariales

LABORATORIO N° 10

Laravel – Mejora de la Aplicación

CODIGO DEL CURSO:



Alumno(s)		Nota
Grupo		
Ciclo		
Fecha de entrega		

I.- OBJETIVOS:

- Crear funcionalidad para eliminar álbumes y fotos.

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.

MEJORANDO EL PROYECTO

En esta sección corregiremos y mejoraremos el proyecto en la parte visual y de usabilidad del sistema.

1. Abra la vista **app.blade.php**
2. Modifique aproximadamente la línea: **31**, cambie el texto Laravel por “Gestor Imágenes”

```
26 <span class="sr-only">Toggle navigation</span>
27 <span class="icon-bar"></span>
28 <span class="icon-bar"></span>
29 <span class="icon-bar"></span>
30 </button>
31 <a class="navbar-brand" href="#">Gestor Imágenes</a>
32 </div>
33
34 <div class="collapse navbar-collapse" id="bs-example-navbar
```

3. Abra el archivo **mostrar.blade.php** de los álbumes y agregue el código siguiente:

```

25 @if(sizeof($albumes)>0)
26     @foreach($albumes as $index => $album)
27         @if($index%3==0)
28             <div class="row">
29                 @endif
30                 <div class="col-sm-6 col-md-4">
31                     <div class="thumbnail">
32                         <div class="caption">
33                             <h3>{{ $album->nombre }}</h3>
34                             <p>{{ $album->descripcion }}</p>
35                             <p><a href="/validado/fotos?id={{ $album->id }}">Ver Foto</a>
36                             <p><a href="/validado/albumes/actualizar-{{ $album->id }}">Actualizar
37                                 Álbum</a></p>
38                             <form action="/validado/albumes/eliminar-{{ $album->id }}" method="POST">
39                                 <input type="hidden" name="_token" value="{{ csrf_token() }}">
40                                 <input type="hidden" name="id" value="{{ $album->id }}">
41                                 <input class="btn btn-danger" role="button" type="submit" value="Eliminar">
42                             </form>
43                         </div>
44                     </div>
45                 @if(($index+1)%3==0)
46                     </div>
47                 @endif
48             @endforeach
49         @else
50             <div class="alert alert-danger">

```

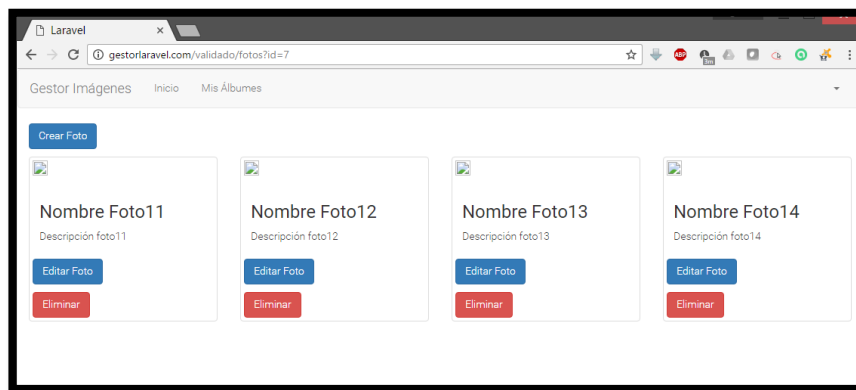
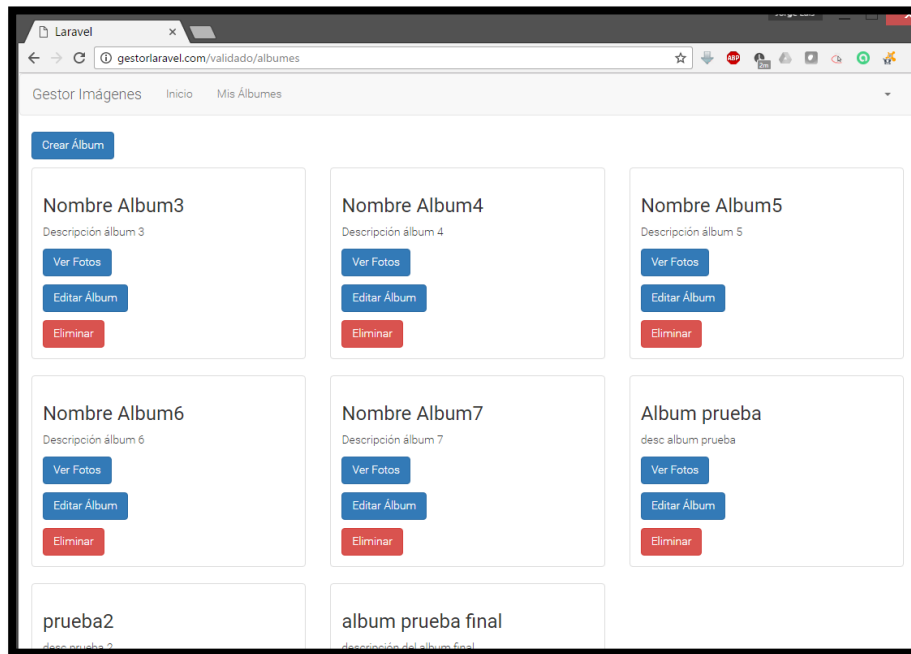
4. Abra el archivo mostrar.blade.php de las fotos y agregue el código siguiente:

```

17 <div class="container-fluid">
18     <p><a href="/validado/fotos/crear-foto?id={{ $id }}" class="btn btn-primary" role="button">Crear
19     Foto</a></p>
20     @if(sizeof($fotos)>0)
21         @foreach($fotos as $index => $foto)
22             @if($index%4==0)
23                 <div class="row">
24                     @endif
25                     <div class="col-sm-6 col-md-3">
26                         <div class="thumbnail">
27                             
28                             <div class="caption">
29                                 <h3>{{ $foto->nombre }}</h3>
30                                 <p>{{ $foto->descripcion }}</p>
31                             </div>
32                             <p><a href="/validado/fotos/actualizar-foto/{{ $foto->id }}" class="btn
33                                 btn-primary" role="button">Editar Foto</a></p>
34                             <form action="/validado/fotos/eliminar-foto" method="POST">
35                                 <input type="hidden" name="_token" value="{{ csrf_token() }}" required>
36                                 <input type="hidden" name="id" value="{{ $foto->id }}" required>
37                                 <input class="btn btn-danger" role="button" type="submit" value="Eliminar">
38                             </form>
39                         </div>
40                     @if(($index+1)%4==0)
41                         </div>
42                     @endif
43                 @endforeach
44             @else
45                 <div class="alert alert-danger">

```

Pruebe el código de modo que, los álbumes, se ordenen en columnas de 3 y las fotos en 4 columnas como máximo.



Sucede también que, al momento de eliminar un álbum de la base de datos, éste elimina todos los registros en la tabla fotos que pertenezcan al álbum en mención, lo cual es correcto. Sin embargo, hemos obviado la parte de eliminar las fotos como archivos en el servidor.

- Modifique el controlador de los álbumes en su acción `postEliminarAlbum`, para que quede de la siguiente manera:

```

60     public function postEliminarAlbum(EliminarAlbumRequest $request){
61
62         $album=Album::find($request->get('id'));
63         $fotos=$album->$fotos;
64         foreach ($fotos as $$foto) {
65             $rutaanterior=getcwd().$foto->$ruta;
66             if (file_exists($rutaanterior)) {
67                 unlink(realpath($rutaanterior));
68             }
69             $foto->delete();
70         }
71         $album->delete();
72         return redirect('/validado/albumes')->with('eliminado','El album fué eliminado!');
73     }

```

Anote los detalles más resaltantes del código anterior:

COMANDO PARA RE-ALIMENTAR LA BASE DE DATOS

1. Para probar el funcionamiento correcto del sistema, nuevamente alimente la base de datos.
2. Abra la ventana de comandos con path en nuestro proyecto.
3. Escriba el comando: **php artisan migrate:refresh --seed**

Es probable que tengan un error del siguiente tipo: (Si no lo ha tenido, omita el paso)

```
C:\wamp64\www\GestorImagenes>php artisan migrate:refresh --seed

[Illuminate\Database\QueryException]
SQLSTATE[42S02]: Base table or view not found: 1051 Unknown table 'gestorim
agenes.users' (SQL: drop table `users`)
```

Si ha tenido el error haga lo siguiente:

1. Abra el archivo de migraciones que le pertenece a la tabla Usuario.
2. Identifique la función down().
3. Dentro de esta función cambie el contenido de la función **drop('users')** por **drop('usuarios')**
4. Si prueba nuevamente el comando el resultado debería ser el siguiente:

```
C:\wamp64\www\GestorImagenes>php artisan migrate:refresh --seed
Rolled back: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table
Migrated: 2016_08_02_222916_CrearTablaAlbumes
Migrated: 2016_08_02_223103_CrearTablaFotos
Seeded: UsuariosSeeder
Seeded: AlbumesSeeder
Seeded: FotosSeeder

C:\wamp64\www\GestorImagenes>
```

5. Acceda con las credenciales creadas recientemente y pruebe la funcionalidad del programa.

OBSERVACIONES Y CONCLUSIONES
