

Projecte GLOBAL: VideosApp

6e SPRINT



Professor: Jordi Vega

Assignatures: M7, M8 i M9

Nom: Marc

Cognoms: Pallares Pino



INDEX

Que s'ha de fer al 6e Sprint?
Link Github

3
21





Que s'ha de fer al 6e Sprint?

Descripció del 6e Sprint:

- Corregir els errors del 5e sprint.
- En cas que al modificar el codi falla algun test d'un sprint anterior, s'han d'arreglar.
- Modificar videos per poder assignar el video a les series.

Creació nou vídeo

Títol

Descripció

URL

Sèrie

Volver

Crear

```
Series.php x  php 2025_03_27_145406_create_series_table.php  VideosManageController.php  Controller.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  4 usages new *
8  class Series extends Model
9  {
10     // Aquí puedes definir las propiedades y métodos de la clase Series
11 }
```

```
no usages new *
public function series(): \Illuminate\Database\Eloquent\Relations\BelongsTo
{
    return $this->belongsTo(related: Series::class);
}
```

- Els regular users han de poder crear videos. Per tant a VideoController afegirem les funcions del crud per als regular. A la vista de videos s'haurà de crear els botons per al crud.

```
<style>
  background: #002060;
}
</style>

<div class="container">
  <h1>All Videos</h1>
  <div class="flex justify-center mb-4">
    <a href="{{ route('videos.manage.create') }}" class="btn-create">Create Video</a>
  </div>
  <div class="flex justify-center">
    <table>
      <thead>
        <tr>
          <th>Title</th>
          <th>Description</th>
          <th>Actions</th>
        </tr>
      </thead>
      <tbody>
        @foreach($videos as $video)
          <tr>
            <td>{{ $video->title }}</td>
            <td>{{ $video->description }}</td>
            <td>
              <a href="{{ route('videos.manage.edit', $video->id) }}" class="btn-edit">Edit</a>
              <form action="{{ route('videos.manage.destroy', $video->id) }}" method="POST" style="display:inline;">
                @csrf
                @method('DELETE')
                <button type="submit" class="btn-delete">Delete</button>
              </form>
            </td>
          </tr>
        </tbody>
      </table>
    </div>
  </div>
```

- Crear migracio series. Ha de tindre els camps id, title, description, image, user_name, user_photo_url, published_at.

```
no usages 1 «Marc»*
7 class CreateSeriesTable extends Migration
8
9 /**
10  * Run the migrations.
11  *
12  * @return void
13  */
14 1 «Marc»*
15 public function up()
16 {
17     Schema::create( table: 'series', function (Blueprint $table) {
18         $table->id();
19         $table->string( column: 'title');
20         $table->text( column: 'description')->nullable();
21         $table->string( column: 'image')->nullable();
22         $table->string( column: 'user_name');
23         $table->string( column: 'user_photo_url')->nullable();
24         $table->timestamp( column: 'published_at')->nullable();
25         $table->timestamps();
26     });
27 }
28
29 /**
30  * Reverse the migrations.
31  *
32  * @return void
33  */
34 1 «Marc»*
35 public function down()
```

- Crear el model de series. Ha de tindre les funcions testedby, videos (per fer la relació 1:N), getFormattedCreatedAtAttribute, getFormattedForHumansCreatedAtAttribute, getCreatedAtTimestampAttribute.

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Carbon\Carbon;
8
9 4 usages 1 «Marc» *
10 class Series extends Model
11 {
12     use HasFactory;
13
14     no usages
15     protected $fillable = [
16         'title', 'description', 'image', 'user_name', 'user_photo_url', 'published_at'
17     ];
18
19     /**
20      * Relación 1:N con el modelo Video.
21      */
22     new *
23     public function videos()
24     {
25         return $this->hasMany([related: Video::class]);
26     }
27
28     /**
29      * Relación con los usuarios que han probado la serie.
30      */
31
32     no usages 1 «Marc» *
```

- Al model de videos afegir la relació 1:N

```

10 class Video extends Model
11 {
12     use HasFactory;
13
14     no usages
15     protected $fillable = [
16         'title', 'description', 'url', 'user_id', 'series_id'
17     ];
18
19     no usages
20     protected $dates = ['published_at'];
21
22     /**
23      * Relación inversa 1:N con el modelo Series.
24      */
25     no usages new *
26     public function series()
27     {
28         return $this->belongsTo(related: Series::class);
29     }
30
31     /**
32      * Obtener la fecha de publicación formateada.
33      */
34     no usages
35     public function getFormattedPublishedAtAttribute()
36     {
37         return $this->published_at ? Carbon::parse($this->published_at)->format('F j, Y') : 'Not Published';
38     }
39

```

- Crear SeriesManageController en les funcions testedby, index, store, edit, update, delete i destroy.

Project: VideosAppmarc

Controllers:

- Controller.php
- SeriesManageController.php
- UsersController.php
- UsersManageController.php
- VideosController.php
- VideosManageController.php

Models:

- Membership.php
- Series.php
- Team.php
- TeamInvitation.php
- User.php
- Video.php

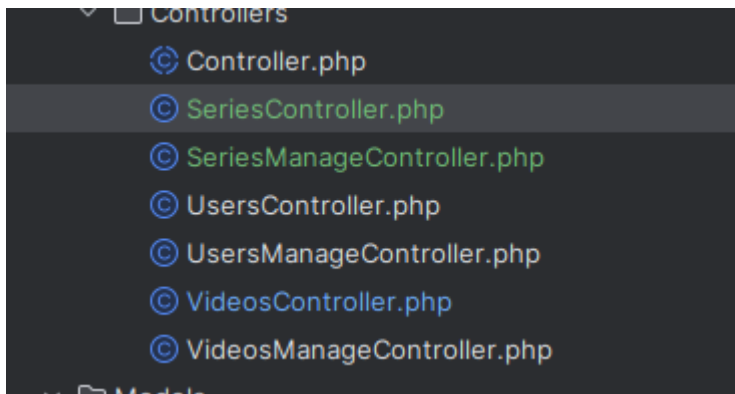
SeriesManageController.php

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Series;
6 use Illuminate\Http\Request;
7
8 no usages new *
9 class SeriesManageController extends Controller
10 {
11     /**
12      * Mostrar una lista de todas las series.
13      */
14     new *
15     public function index()
16     {
17         $series = Series::all();
18         return view('series.index', compact('series'));
19     }
20
21     /**
22      * Mostrar el formulario para crear una nueva serie.
23      */
24     new *
25     public function create()
26     {
27         return view('series.create');
28     }
29
30     /**
31      * Almacenar una nueva serie en la base de datos.
32      */
33

```


- Crear SeriesController en les funcions index i show.



```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Series;
6 use Illuminate\Http\Request;
7
8 /**
9  *
10  */
11 class SeriesController extends Controller
12 {
13     /**
14      * Mostrar una lista de todas las series.
15      */
16     new *
17     public function index()
18     {
19         $series = Series::all();
20         return view('series.index', compact('var_name' => 'series'));
21     }
22
23     /**
24      * Mostrar una serie especifica.
25      */
26     new *
27     public function show(Series $series)
28     {
29         return view('series.show', compact('var_name' => 'series'));
30     }
31 }
```

- Crear el model de Serie amb les funcions testedby, videos, getFormattedCreatedAtAttribute(), getFormattedForHumansCreatedAtAttribute() i getCreatedAtTimestampAttribute().

```
© SeriesManageController.php  © SeriesController.php  © Series.php x
9      class Series extends Model
23     }
24
25     /**
26      * Relación con los usuarios que han probado la serie.
27      */
28     no usages new *
29     public function testedBy()
30     {
31         return $this->belongsToMany(related: User::class, table: 'series_user_tested');
32     }
33
34     /**
35      * Obtener la fecha de creación formateada.
36      */
37     no usages new *
38     public function getFormattedCreatedAtAttribute()
39     {
40         return $this->created_at->format('d/m/Y');
41     }
42
43     /**
44      * Obtener la fecha de creación en formato humano.
45      */
46     no usages new *
47     public function getFormattedForHumansCreatedAtAttribute()
48     {
49         return $this->created_at->diffForHumans();
50     }
51 }
```

- Crear la migració de la serie ha de tenir els camps, id, title, description, image (nullable), user_name, user_photo_url (nullable), published_at (nullable).

```
© SeriesManageController.php © SeriesController.php © Series.php php 2025_03_27_145406_create_s
2
3 > use ...
6
no usages  ⓘ «Marc» *
7 class CreateSeriesTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     ⓘ «Marc» *
15     public function up()
16     {
17         Schema::create(table: 'series', function (Blueprint $table) {
18             $table->id();
19             $table->string(column: 'title');
20             $table->text(column: 'description')->nullable();
21             $table->string(column: 'image')->nullable();
22             $table->string(column: 'user_name');
23             $table->string(column: 'user_photo_url')->nullable();
24             $table->timestamp(column: 'published_at')->nullable();
25             $table->timestamps();
26         });
27     }
28     /**
29      * Reverse the migrations.
30      *
31      * @return void
```

- A helpers crear la funció `create_series()` ha d'haver 3 series.

```
esController.php  © Series.php  © DefaultVideos.php  ×  show.blade.php  © User.php  © U
9  class DefaultVideos
56
57  1 usage new *
58  public static function create_series()
59  {
60      $seriesData = [
61          [
62              'title' => 'Series 1',
63              'description' => 'Description for Series 1',
64              'image' => 'image1.jpg',
65              'user_name' => 'User 1',
66              'user_photo_url' => 'user1.jpg',
67              'published_at' => now(),
68          ],
69          [
70              'title' => 'Series 2',
71              'description' => 'Description for Series 2',
72              'image' => 'image2.jpg',
73              'user_name' => 'User 2',
74              'user_photo_url' => 'user2.jpg',
75              'published_at' => now(),
76          ],
77          [
78              'title' => 'Series 3',
79              'description' => 'Description for Series 3',
80              'image' => 'image3.jpg',
81              'user_name' => 'User 3',
82              'user_photo_url' => 'user3.jpg',
83              'published_at' => now(),
84          ],
85      ],
86  }
```

Creació nou vídeo

Títol

a

Descripció

a

URL

https://www.youtube.com/watch?v=tT9Eh8wNMkw&ab_c

Sèrie

Series 2

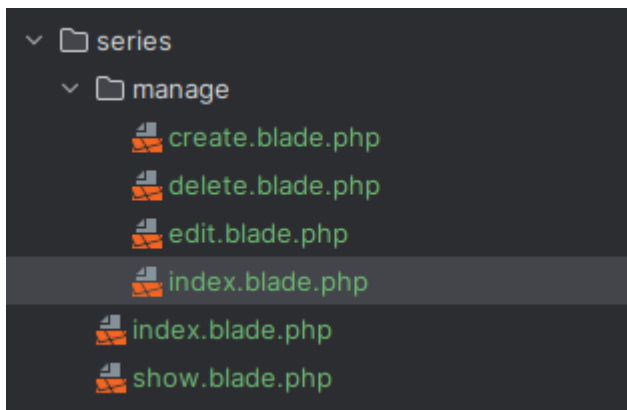
Selecciona una sèrie

Series 1

Series 2

Series 3

- Crear les vistes per al CRUD que només poden veure-ho els que tinguin els permisos adients: `resources/views/series/manage/index.blade.php`, `resources/views/series/manage/create.blade.php`, `resources/views/series/manage/edit.blade.php`, `resources/views/series/manage/delete.blade.php`.



- A la vista `index.blade.php`, afegir la taula del CRUD de series.

```
index.blade.php x
1  @extends('layouts.app')
2
3  @section('content')
4      @can('view series')
5          <div class="container mx-auto mt-8">
6              <h1 class="text-2xl font-bold mb-4">All Series</h1>
7              <div class="flex justify-center mb-4">
8                  <a href="{{ route('series.manage.create') }}" class="btn-create">Create Series</a>
9              </div>
10             <div class="flex justify-center">
11                 <table>
12                     <thead>
13                         <tr>
14                             <th>Title</th>
15                             <th>Description</th>
16                             <th>Actions</th>
17                         </tr>
18                     </thead>
19                     <tbody>
20                         @foreach($series as $serie)
21                             <tr>
22                                 <td>{{ $serie->title }}</td>
23                                 <td>{{ $serie->description }}</td>
24                                 <td>
25                                     <a href="{{ route('series.manage.edit', $serie->id) }}" class="btn-edit">Edit</a>
26                                     <form action="{{ route('series.manage.destroy', $serie->id) }}" method="POST" style="display: inline-block;">
27                                         @csrf
28                                         @method('DELETE')
29                                         <button type="submit" class="btn-delete">Delete</button>
30                                     </form>

```

- A la vista create.blade.php, afegir el formulari per posar les series, s'ha d'utilitzar l'atribut data-qa per a que sigui més fàcil identificar per als tests.

```

create.blade.php x
1 @extends('layouts.app')
2
3 @section('content')
4     @can('create series')
5         <div class="container mx-auto mt-8">
6             <h1 class="text-2xl font-bold mb-4">Create Series</h1>
7             <form action="{{ route('series.manage.store') }}" method="POST" data-qa="create-series-form">
8                 @csrf
9                 <div class="mb-4">
10                     <label for="title" class="block text-white data-qa="title-label">Title</label>
11                     <input type="text" name="title" id="title" class="form-control border border-gray-300 text-black p-2 w-full">
12                 </div>
13                 <div class="mb-4">
14                     <label for="description" class="block text-white data-qa="description-label">Description</label>
15                     <textarea name="description" id="description" class="form-control border border-gray-300 text-black p-2 w-full">
16                 </div>
17                 <button type="submit" class="text-white font-bold py-2 px-4 rounded" data-qa="submit-button">Create</button>
18             </form>
19         </div>
20     @else
21         <p>You do not have permission to create a series.</p>
22     @endcan
23 @endsection

```

- A la vista edit.blade.php, afegir la taula del CRUD de series.

```

edit.blade.php x
1 @extends('layouts.app')
2
3 @section('content')
4     @can('edit series')
5         <div class="container mx-auto mt-8">
6             <h1 class="text-2xl font-bold mb-4">Edit Series</h1>
7             <form action="{{ route('series.manage.update', $series->id) }}" method="POST">
8                 @csrf
9                 @method('PUT')
10                <div class="mb-4">
11                    <label for="title" class="block text-white">Title</label>
12                    <input type="text" name="title" id="title" class="form-control border border-gray-300 text-black p-2 w-full">
13                </div>
14                <div class="mb-4">
15                    <label for="description" class="block text-white">Description</label>
16                    <textarea name="description" id="description" class="form-control border border-gray-300 text-black p-2 w-full">
17                </div>
18                <button type="submit" class="text-white font-bold py-2 px-4 rounded">Update</button>
19            </form>
20
21            <div class="flex justify-center mt-8">
22                <table>
23                    <thead>
24                        <tr>
25                            <th>Title</th>
26                            <th>Description</th>
27                            <th>Actions</th>
28                        </tr>
29                    </thead>
30                    <tbody>

```

- A la vista delete.blade.php, afegir la confirmació de l'eliminació de les sèries i els videos associats a la serie. (Si no es vol borrar els videos es pot fer que es desassigni la relació).

```
1 @extends('layouts.app')
2
3 @section('content')
4     @can('delete series')
5         <div class="container mx-auto mt-8">
6             <h1 class="text-2xl font-bold mb-4">Delete Series</h1>
7             <p>Are you sure you want to delete the series titled "{{ $serie->title }}"?</p>
8             <form action="{{ route('series.manage.destroy', $serie->id) }}" method="POST">
9                 @csrf
10                 @method('DELETE')
11                 <div class="mb-4">
12                     <input type="checkbox" name="delete_videos" id="delete_videos" value="1">
13                     <label for="delete_videos" class="text-white">Also delete associated videos</label>
14                 </div>
15                 <button type="submit" class="bg-red-600 hover:bg-red-700 text-white font-bold py-2 px-4 rounded">Yes, Delete</button>
16                 <a href="{{ route('series.manage.index') }}" class="bg-gray-500 hover:bg-gray-700 text-white font-bold py-2 px-4 rounded">Cancel</a>
17             </form>
18         </div>
19     @else
20         <p>You do not have permission to delete this series.</p>
21     @endcan
22 @endsection
```

- Crear la vista de resources/views/series/index.blade.php on es vegin totes les sèries i es puguin buscar i al clicar a la serie que mostri els videos que té aquella serie.

```
1 @extends('layouts.app')
2
3 @section('content')
4     @can('view series')
5         <div class="container mx-auto mt-8">
6             <h1 class="text-2xl font-bold mb-4">All Series</h1>
7             <div class="flex justify-center mb-4">
8                 <form action="{{ route('series.index') }}" method="GET" class="w-full max-w-sm">
9                     <input type="text" name="search" placeholder="Search series..." class="form-control border border-gray-300 text-black p-2 w-full">
10                     <button type="submit" class="btn-search mt-2">Search</button>
11                 </form>
12             </div>
13             <div class="flex justify-center">
14                 <table>
15                     <thead>
16                         <tr>
17                             <th>Title</th>
18                             <th>Description</th>
19                             <th>Actions</th>
20                         </tr>
21                     </thead>
22                     <tbody>
23                         @foreach($series as $serie)
24                             <tr>
25                                 <td>{{ $serie->title }}</td>
26                                 <td>{{ $serie->description }}</td>
27                                 <td>
28                                     <a href="{{ route('series.show', $serie->id) }}" class="btn-view">View Videos</a>
29                                 </td>
30                             </tr>
31                         @endforeach
32                     </tbody>
33                 </table>
34             </div>
35         </div>
36     @endcan
37 @endsection
```


- A helpers crear els permisos de gestió de les series per al crud i assignar-los als usuaris superadmin.

```
1 usage new *
51 public static function create_series_management_permission()
52 {
53     $permission = 'manageSeries';
54
55     if (!Permission::where('name', $permission)->exists()) {
56         Permission::firstOrCreate(['name' => $permission]);
57     }
58
59     $superAdminRole = Role::firstOrCreate(['name' => 'super-admin', 'guard_name' => 'web']);
60     if (!$superAdminRole->hasPermissionTo($permission)) {
61         $superAdminRole->givePermissionTo($permission);
62     }
63 }
64
65 1 usage new *
66 private static function create_series_management_permissions()
67 {
68 }
```

- A test/Unit/SerieTest crear la funció:
serie_have_videos()

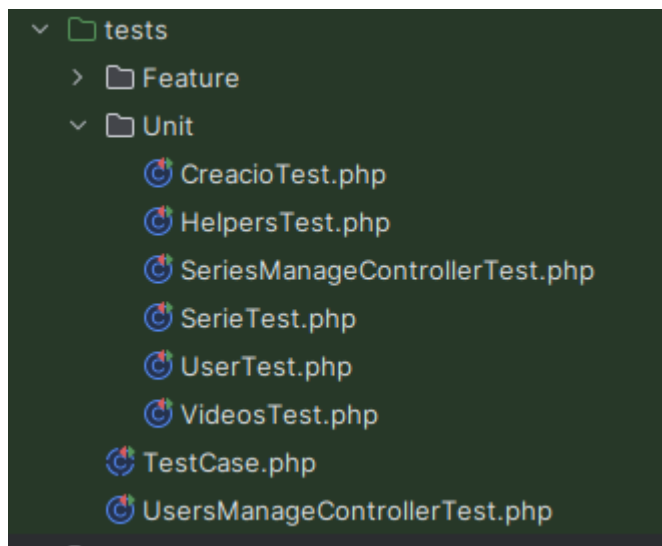
```
✓ Tests passed: 1 of 1 test - 20ms

/bin/php -dxdebug.remote_enable=1 -dxdebug.remote_mode=req -dxdebug.remote.
Testing started at 23:26 ...
Testing started at 23:26 ...
PHPUnit 11.4.4 by Sebastian Bergmann and contributors.

Runtime:      PHP 8.3.6
Configuration: /home/alumnat/Escriptori/2n/mp09/laravelmarc/VideosAppmarc/
Time: 00:00.226, Memory: 40.50 MB

OK, but there were issues!
Tests: 1, Assertions: 2, PHPUnit Deprecations: 1.
Process finished with exit code 0
```

- A SeriesManageControllerTest crear les funcions:
loginAsVideoManager,
loginAsSuperAdmin,
loginAsRegularUser,
user_with_permissions_can_see_add_series,
user_without_series_manage_create_cannot_see_add_series,
user_with_permissions_can_store_series,
user_without_permissions_cannot_store_series,
user_with_permissions_can_destroy_series,
user_without_permissions_cannot_destroy_series,
user_with_permissions_can_see_edit_series,
user_without_permissions_cannot_see_edit_series,
user_with_permissions_can_update_series,
user_without_permissions_cannot_update_series,
user_with_permissions_can_manage_series,
regular_users_cannot_manage_series,
guest_users_cannot_manage_series,
videomanagers_can_manage_series,
superadmins_can_manage_series



- Crear les rutes de series/manage per al CRUD de les series amb el seu middleware corresponent i la ruta de l'índex i el show de series. Les rutes del CRUD i les de l'índex i show han d'aparèixer només quan estàs logejat.

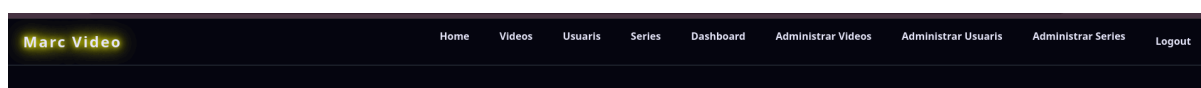
```
Route::middleware(['auth', 'can:manageseries'])->group(function () {
    Route::get(uri: '/series/manage', [SeriesManageController::class, 'index'])->name(name: 'series.manage.index');
    Route::get(uri: '/series/manage/create', [SeriesManageController::class, 'create'])->name(name: 'series.manage.create');
    Route::post(uri: '/series/manage', [SeriesManageController::class, 'store'])->name(name: 'series.manage.store');
    Route::get(uri: '/series/manage/{id}/edit', [SeriesManageController::class, 'edit'])->name(name: 'series.manage.edit');
    Route::put(uri: '/series/manage/{id}', [SeriesManageController::class, 'update'])->name(name: 'series.manage.update');
    Route::delete(uri: '/series/manage/{id}', [SeriesManageController::class, 'destroy'])->name(name: 'series.manage.destroy');
});

Route::get(uri: '/users', [UsersController::class, 'index'])->name(name: 'users.index');
Route::get(uri: '/users/{id}', [UsersController::class, 'show'])->name(name: 'users.show');
});

Route::get(uri: '/videos', [VideosController::class, 'index'])->name(name: 'videos.index');
Route::get(uri: '/videos/{id}', [VideosController::class, 'show'])->name(name: 'videos.show');

Route::get(uri: '/series', [SeriesController::class, 'index'])->name(name: 'series.index');
Route::get(uri: '/series/{id}', [SeriesController::class, 'show'])->name(name: 'series.show');
```

- S'ha de poder navegar entre pàgines.



- Afegir a resources/markdown/terms el que heu fet al sprint.

```
## Sprint 6
**Modificar videos per poder assignar el video a les series**.
**Els regular users han de poder crear videos**. Per tant a VideoController afegirem les funcions del crud
**Crear migracio series**. Ha de tindre els camps id, title, description, image, user\_name, user\_photo\_u
**Crear el model de series**. Ha de tindre les funcions testedby, videos (per fer la relació 1:N), getForma
**Al model de videos afegir la relació 1:N**.
**Crear SeriesManageController** en les funcions testedby, index, store, edit, update, delete i destroy.
**Crear SeriesController** en les funcions index i show.
**Crear el model de Serie** amb les funcions testedby, videos, getFormattedCreatedAtAttribute(), getFormatt
**Crear la migració de la serie** ha de tenir els camps, id, title, description, image (nullable), user\_na
**A helpers crear la funció create\_series()** ha d'haver 3 series.
**Crear les vistes per al CRUD** que només poden veure-ho els que tinguin els permisos adients: 'resources/
**A la vista index.blade.php**, afegir la taula del CRUD de series.
**A la vista create.blade.php**, afegir el formulari per posar les series, s'ha d'utilitzar l'atribut 'data
**A la vista edit.blade.php**, afegir la taula del CRUD de series.
**A la vista delete.blade.php**, afegir la confirmació de l'eliminació de les sèries i els videos associats
**Crear la vista de resources/views/series/index.blade.php** on es vegin totes les sèries i es puguin busca
**A helpers crear els permisos de gestió de les series** per al crud i assignar-los als usuaris superadmin.
**A test/Unit/SerieTest crear la funció**:
- 'serie\_have\_videos()'

**A SeriesManageControllerTest crear les funcions**:
- 'loginAsVideoManager'
- 'loginAsSuperAdmin'
- 'loginAsRegularUser'
- 'user\_with\_permissions\_can\_see\_add\_series'
- 'user\_without\_series\_manage\_create\_cannot\_see\_add\_series'
- 'user\_with\_permissions\_can\_store\_series'
- 'user\_without\_permissions\_cannot\_store\_series'
- 'user\_with\_permissions\_can\_destroy\_series'
- 'user\_without\_permissions\_cannot\_destroy\_series'
```

Comprovar en Larastan: Comprovar en Larastan tots els fitxers que heu creat.

Sprint 6

Modificar videos per poder assignar el video a les series. Els regular users han de poder crear videos. Per tant a VideoController afegirem les funcions del crud per als regular. A la vista de videos s'haurà de crear els botons per al crud.


Crear migracio series. Ha de tindre els camps id, title, description, image, user_name, user_photo_url, published_at. **Crear el model de series.** Ha de tindre les funcions testedby, videos (per fer la relació 1:N), getFormattedCreatedAtAttribute(), getFormattedForHumansCreatedAtAttribute(), getCreatedAtTimestampAttribute(). **Al model de videos afegir la relació 1:N.** **Crear SeriesManageController** en les funcions testedby, index, store, edit, update, delete i destroy. **Crear SeriesController** en les funcions index i show. **Crear el model de Serie** amb les funcions testedby, videos, getFormattedCreatedAtAttribute(), getFormattedForHumansCreatedAtAttribute() i getCreatedAtTimestampAttribute(). **Crear la migració de la serie** ha de tenir els camps, id, title, description, image (nullable), user_name, user_photo_url (nullable), published_at (nullable). **A helpers crear la funció create_series()** ha d'haver 3 series. **Crear les vistes per al CRUD** que només poden veure-ho els que tinguin els permisos adients:

resources/views/series/manage/index.blade.php, resources/views/series/manage/create.blade.php, resources/views/series/manage/edit.blade.php, resources/views/series/manage/delete.blade.php. **A la vista index.blade.php,** afegir la taula del CRUD de series. **A la vista create.blade.php,** afegir el formulari per posar les series, s'ha d'utilitzar l'atribut 'data-qa' per a que sigui més facil identificar per als tests. **A la vista edit.blade.php,** afegir la taula del CRUD de series. **A la vista delete.blade.php,** afegir la confirmació de l'eliminació de les sèries i els videos associats a la serie. (Si no es vol borrar els videos es pot fer que es desassigni la relació). **Crear la**

- Comprovar en Larastan tots els fitxers que heu creat.

```
Line  app/Http/Controllers/SeriesController.php
-----
24  Relation 'videos' is not found in App\Models\Serie model.
    larastan.relationExistence
-----

[ERROR] Found 1 error
```

 INSTITUT DE L'EBRE TORTOSA	GLOBAL M7, M8 i M9	DAM 2n
--	--------------------	-----------

Link Github

<https://github.com/Mpallares1/laravelmarc>