

Projecte GLOBAL: VideosApp

3r SPRINT



Professor: Jordi Vega

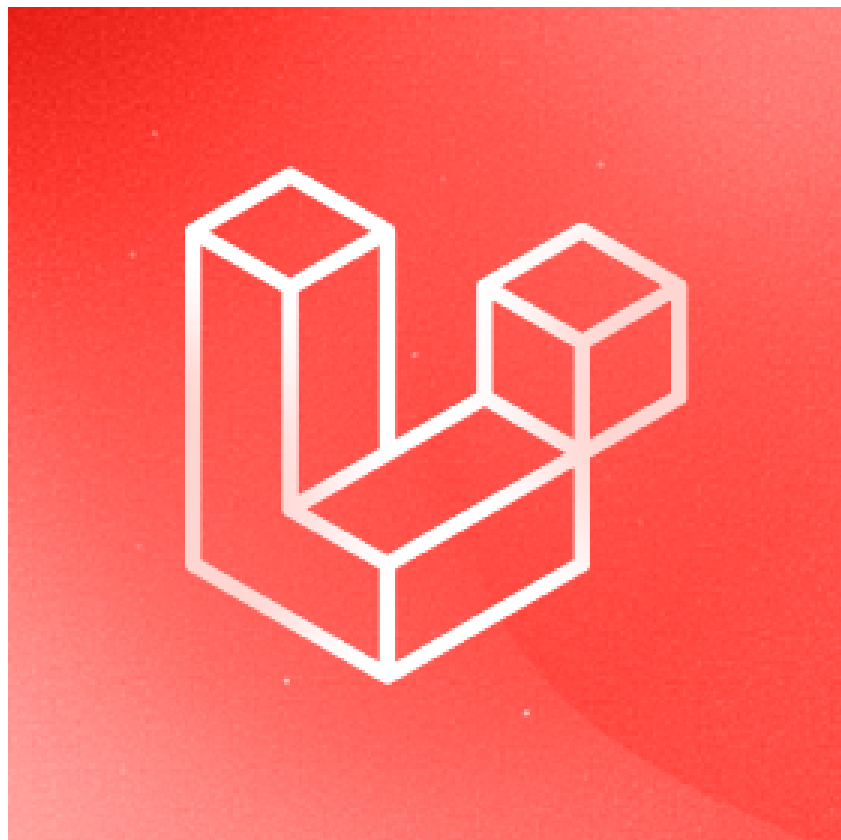
Assignatures: M7, M8 i M9

Nom: Marc

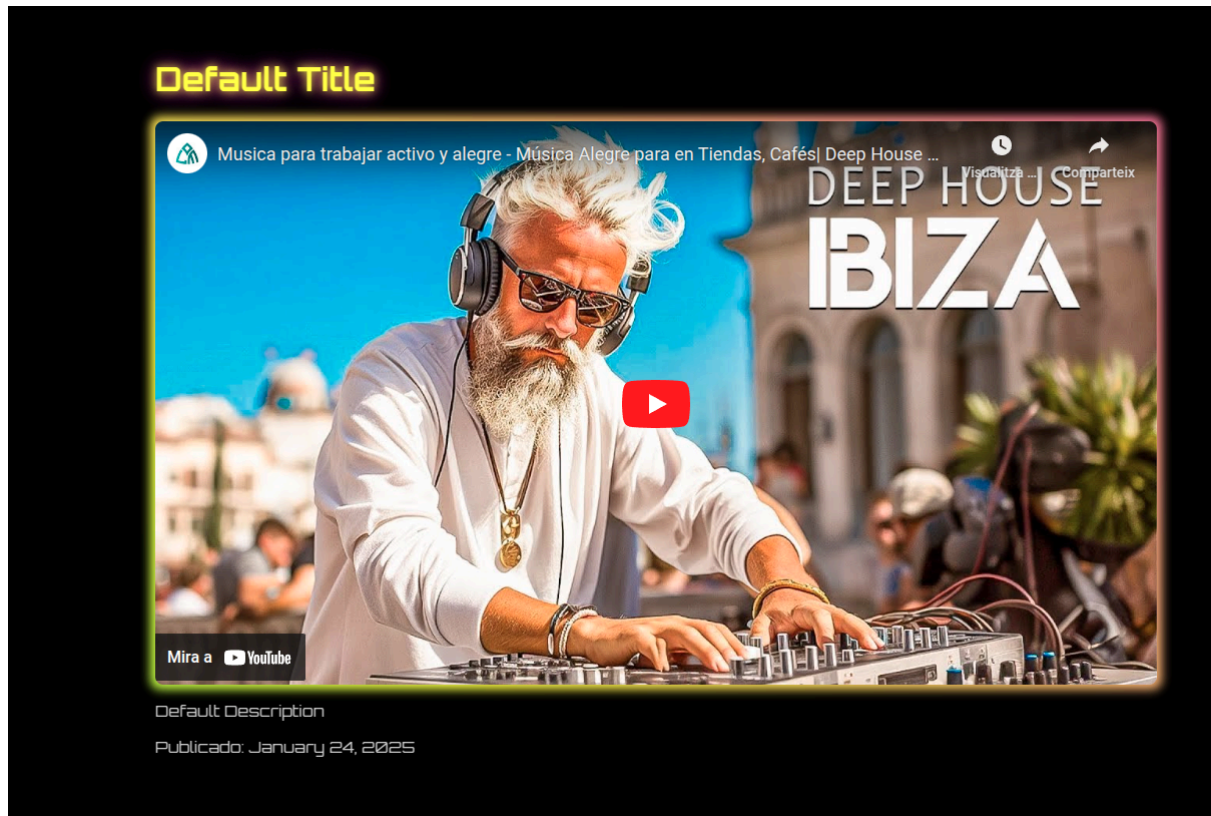
Cognoms: Pallares Pino

Index

• Corregir els errors del 2n sprint.	3
• S'ha d'instal·lar el paquet spatie/laravel-permission. https://spatie.be/docs/laravel-permission/v6/installation-laravel	4
• S'ha de crear una migració per afegir el camp super_admin a la taula dels usuaris.	5
• Model d'usuaris. S'ha d'afegir la funció testedBy() i isSuperAdmin().	5
• Afegir el superadmin al professor a la funció del create_default_professor de helpers.	6
• A app/Providers/AppServiceProvider a la funció de book s'ha de registrar les polítiques d'autorització i també s'han de definir les portes d'accés.	7
• Posar els permisos i els usuaris superadmin, regular user i video manager per defecte al DatabaseSeeder.	8
• Publicar els stubs. Exemple a https://laravel-news.com/customizing-stubs-in-laravel	8
• Crear el test VideosManageControllerTest a la carpeta tests/ Feature/Videos.	9
• Crear el test UserTest a la carpeta tests/Unit. S'han de crear la funció isSuperAdmin() .	10
• Afegir a resources/markdown/terms el que heu fet al sprint.	11
• Comprovar en Larastan tots els fitxers que heu creat.	12



- Corregir els errors del 2n sprint.



```
<p class="mt-2 text-sm" style="color: #ccc;">
  Publicado: {{ $video->published_at->format('F j, Y') }}
</p>
</div>
@endsection
```

- S'ha d'instal·lar el paquet spatie/laravel-permission.

<https://spatie.be/docs/laravel-permission/v6/installation-laravel>

```
alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ composer require spatie/laravel-permission
./composer.json has been updated
Running composer update spatie/laravel-permission
Loading composer repositories with package information
```

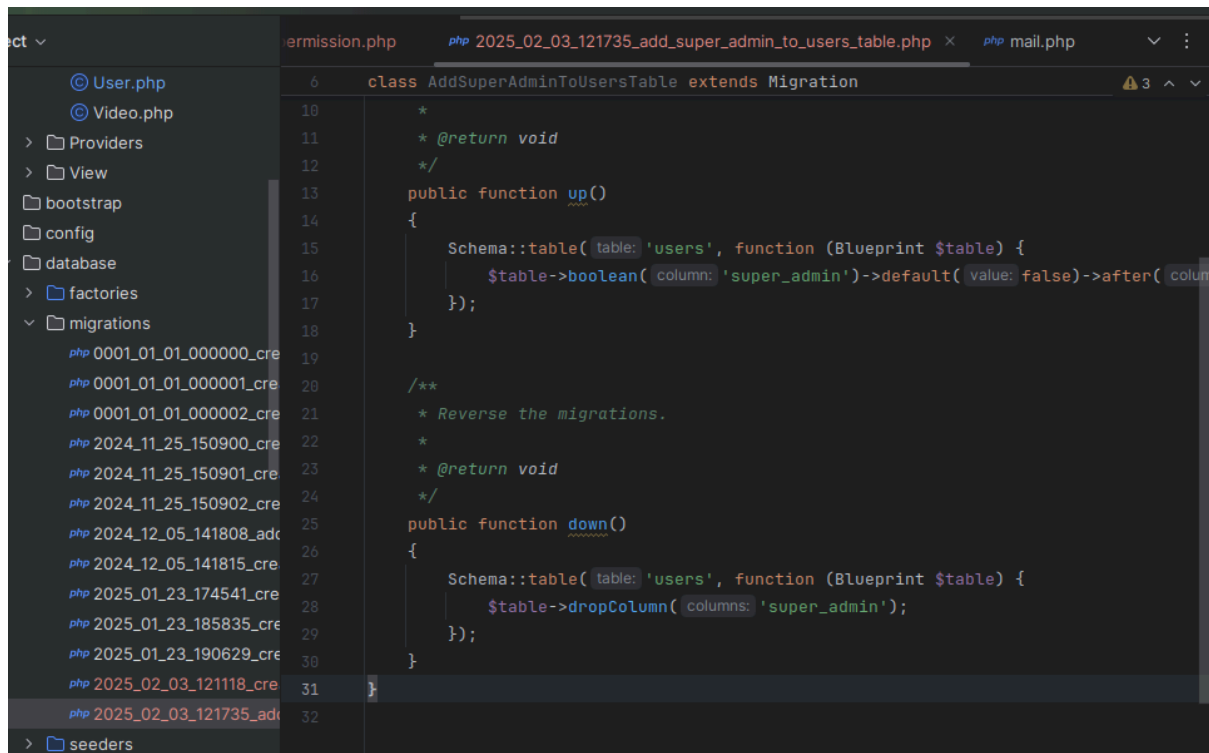
```
alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ php artisan vendor:publish --provider="Spatie\Permission\PermissionServiceProvider"
```

```

16     ],
17
18     'column_names' => [
19         'role_pivot_key' => null,
20         'permission_pivot_key' => null,
21         'model_morph_key' => 'model_id',
22         'team_foreign_key' => 'team_id',
23     ],
24
25     'register_permission_check_method' => true,
26     'register_octane_reset_listener' => false,
27     'teams' => false,
28     'team_resolver' => \Spatie\Permission\DefaultTeamResolver::class,
29     'use_passport_client_credentials' => false,
30     'display_permission_in_exception' => false,
31     'display_role_in_exception' => false,
32     'enable_wildcard_permission' => false,
33
34     'cache' => [
35         'expiration_time' => \DateInterval::createFromDateString('48 hours'),
36         'key' => 'spatie.permission.cache',
37         'store' => 'default',
38     ],
39 ];
40

```

- S'ha de crear una migració per afegir el camp `super_admin` a la taula dels usuaris.

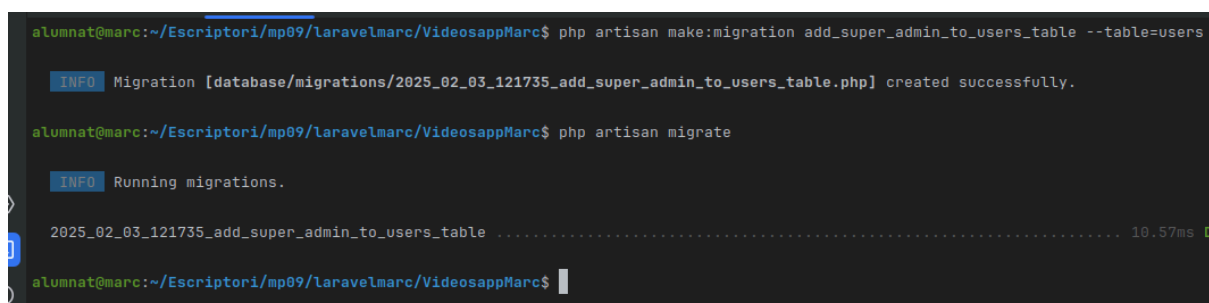


```

class AddSuperAdminToUsersTable extends Migration
{
    /**
     * @return void
     */
    public function up()
    {
        Schema::table('users', function (Blueprint $table) {
            $table->boolean('super_admin')->default('false')->after('column');
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::table('users', function (Blueprint $table) {
            $table->dropColumn('super_admin');
        });
    }
}

```



```

alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ php artisan make:migration add_super_admin_to_users_table --table=users
INFO Migration [database/migrations/2025_02_03_121735_add_super_admin_to_users_table.php] created successfully.

alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ php artisan migrate
INFO Running migrations.

2025_02_03_121735_add_super_admin_to_users_table ..... 10.57ms 0

alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$

```

- Model d'usuaris. S'ha d'afegir la funció `testedBy()` i `isSuperAdmin()`.

```
no usages new *
public function testedBy(): HasMany
{
    return $this->hasMany(related: Test::class, foreignKey: 'user_id');
}

/**
 * Verifica si el usuario es un super administrador.
 *
 * @return bool
 */
no usages new *
public function isSuperAdmin(): bool
{
    return $this->super_admin;
}
```

- Afegir el superadmin al professor a la funció del create_default_professor de helpers.

```

17 class CreateUsers
22     public static function creacioUsuariDefecte(array $user)
30         return DB::transaction(function () use ($user) {
38             // ...
39         });
40     }
41
42     2 usages ± «Marc» *
43     protected static function creacioTeam(User $user): void
44     {
45         // Create a team for the user
46         $user->ownedTeams()->create([
47             'name' => "{$user->name}'s Team",
48             'personal_team' => true, // Ensure the personal_team field is set
49         ]);
50     }
51
52     no usages new *
53     public static function createRegularUser()
54     {
55         return self::createUser([
56             'name' => 'Regular',
57             'email' => 'regular@videosapp.com',
58             'password' => '123456789',
59         ]);
60     }

```

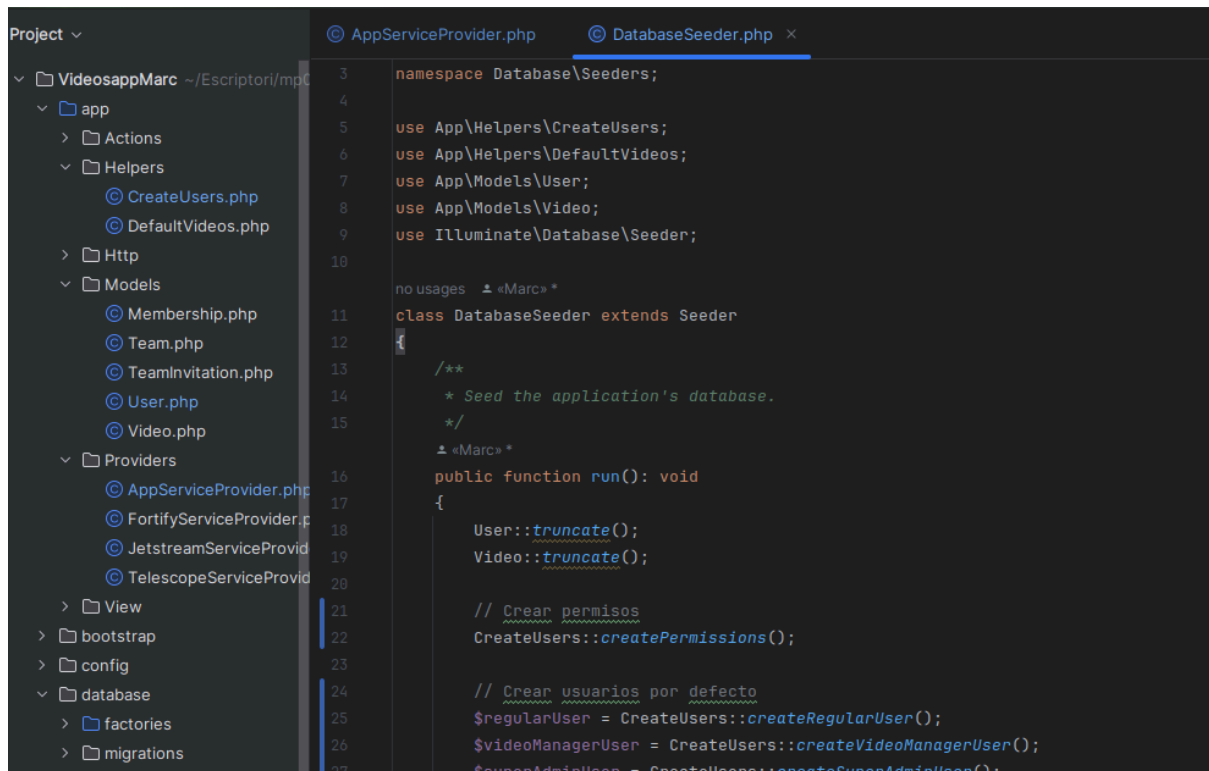
- A app/Providers/AppServiceProvider a la funció de book s'ha de registrar les polítiques d'autorització i també s'han de definir les portes d'accés.

```

1 <?php
2
3 namespace App\Providers;
4
5 use Illuminate\Support\Facades\Gate;
6 use Illuminate\Foundation\Support\Providers\AuthServiceProvider as ServiceProvider;
7 use App\Models\Book;
8 use App\Policies\BookPolicy;
9
10 3 usages ± «Marc» *
11 class AppServiceProvider extends ServiceProvider
12 {
13     /**
14      * The policy mappings for the application.
15      *
16      * @var array
17      */
18     no usages
19     protected $policies = [
20         Book::class => BookPolicy::class,
21     ];
22
23     /**
24      * Register any application services.
25      *
26      * @return void
27      */

```

- Posar els permisos i els usuaris superadmin, regular user i video manager per defecte al DatabaseSeeder.



```

3 namespace Database\Seeders;
4
5 use App\Helpers\CreateUsers;
6 use App\Helpers\DefaultVideos;
7 use App\Models\User;
8 use App\Models\Video;
9 use Illuminate\Database\Seeder;
10
11 no usages 1 «Marc» *
12 class DatabaseSeeder extends Seeder
13 {
14     /**
15      * Seed the application's database.
16      */
17     1 «Marc» *
18     public function run(): void
19     {
20         User::truncate();
21         Video::truncate();
22
23         // Crear permisos
24         CreateUsers::createPermissions();
25
26         // Crear usuarios por defecto
27         $regularUser = CreateUsers::createRegularUser();
28         $videoManagerUser = CreateUsers::createVideoManagerUser();
29         $superAdminUser = CreateUsers::createSuperAdminUser();

```

- Publicar els stubs. Exemple a <https://laravel-news.com/customizing-stubs-in-laravel>

```

no hi ha res a cometre, l'arbre de treball està net
alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ php artisan stub:publish

```

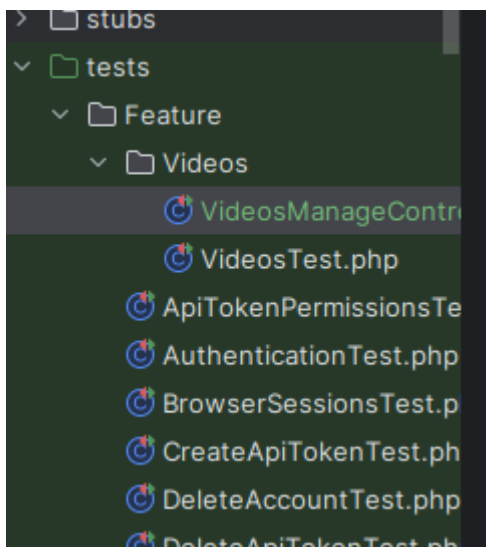


```
no hi ha res a cometre, l'arbre de treball esta net  
alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ git add stubs
```

- Crear el test VideosManageControllerTest a la carpeta tests/ Feature/Videos.

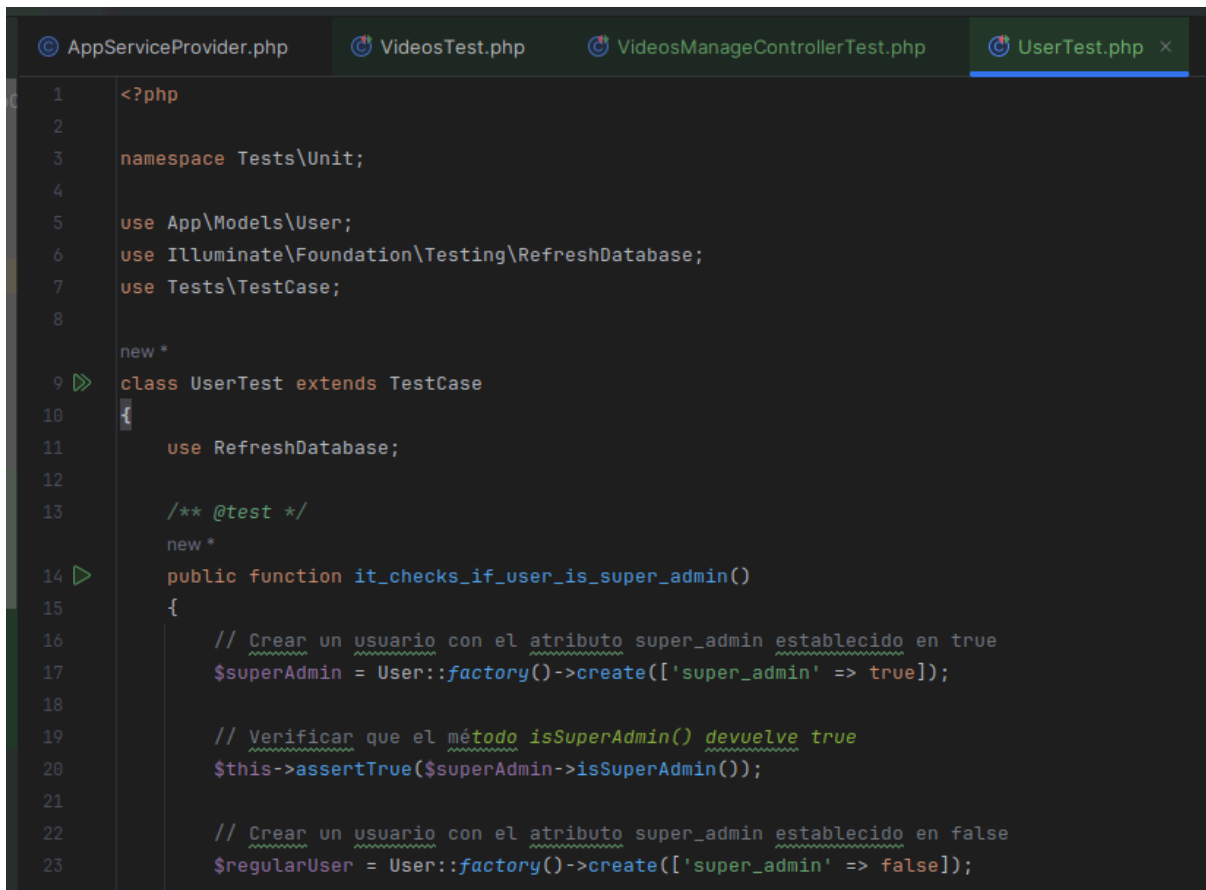
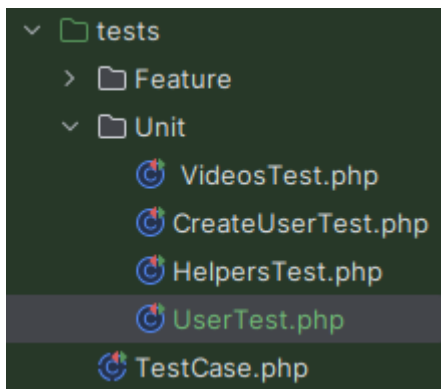
Crear les funcions per comprovar la formatació del video:

user_with_permissions_can_manage_videos(),
regular_users_cannot_manage_videos(),
guest_users_cannot_manage_videos(),
superadmins_can_manage_videos(), loginAsVideoManager(),
loginAsSuperAdmin(), loginAsRegularUser().



```
AppServiceProvider.php  VideosTest.php  VideosManageControllerTest.php x
1  <?php
2
3  namespace Tests\Feature\Videos;
4
5  use App\Models\User;
6  use Illuminate\Foundation\Testing\RefreshDatabase;
7  use Tests\TestCase;
8
9  new *
10 class VideosManageControllerTest extends TestCase
11 {
12     use RefreshDatabase;
13
14     /** @test */
15     new *
16     public function user_with_permissions_can_manage_videos()
17     {
18         $this->loginAsVideoManager();
19
20         $response = $this->get( uri: '/videos' );
21         $response->assertStatus( status: 200 );
22     }
23
24     /** @test */
25     new *
26     public function regular_users_cannot_manage_videos()
```

- Crear el test UserTest a la carpeta tests/Unit. S'han de crear la funció isSuperAdmin() .



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

```

## Sprint 3

En el tercer sprint, hem realitzat les següents tasques:

- **Gestió de Vídeos**:
  - S'ha creat el controlador `VideosManageController`.
  - S'han implementat les proves `VideosManageControllerTest`
  - S'han afegit proves per a la visualització de vídeos en

- **Usuaris i Rols**:
  - S'han creat els rols `Super Admin`, `Video Manager` i `Regular User`.
  - S'han assignat els permisos corresponents a cada rol.
  - S'ha implementat la prova `UserTest` per verificar la funció `isSuperAdmin()`.

- **Personalització de Stubs**:
  - S'han publicat i personalitzat els stubs de Laravel per

```

Sprint 3

En el tercer sprint, hem realitzat les següents tasques:

- **Gestió de Vídeos:**
 - S'ha creat el controlador `VideosManageController`.
 - S'han implementat les proves `VideosManageControllerTest` per verificar els permisos de gestió de vídeos.
 - S'han afegit proves per a la visualització de vídeos en `VideosTest`.
- **Usuaris i Rols:**
 - S'han creat els rols `Super Admin`, `Video Manager` i `Regular User`.
 - S'han assignat els permisos corresponents a cada rol.
 - S'ha implementat la prova `UserTest` per verificar la funció `isSuperAdmin()`.
- **Personalització de Stubs:**
 - S'han publicat i personalitzat els

- Comprovar en Larastan tots els fitxers que heu creat.

```

alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$ vendor/bin/phpstan analyse
Note: Using configuration file /home/alumnat/Escriptori/mp09/laravelmarc/VideosappMarc/phpstan.neon.
46/46 [████████████████████████████████████████] 100%

[OK] No errors

alumnat@marc:~/Escriptori/mp09/laravelmarc/VideosappMarc$

```

Link github:

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