

Projecte IonicVideosApp



Professor: Jordi Vega

Assignatura: M8 Programació multimèdia i dispositius mòbils

Nom: Harvey

Cognoms: John Glover

INDEX:

Instal·lació Inicial	3
Paquets a instalar:	3
Backend (Laravel - VideosApp):	5
Crear el localStorage	5
database/factories/MultimediaFactory.php	5
config/filesystems.php	6
config/cors.php	6
Crear el controller ApiMultimediaController	7
INDEX:	7
STORE:	8
STORE:	9
UPDATE:	10
DESTROY/DELETE:	11
Crear el Model	13
app/Models/Multimedia.php	13
database/migrations/2025_03_24_153958_create_multimedia_table.php	14
Crear la api	15
routes/api.php	15
app/Http/Controllers/AuthController.php	16
app/Http/Controllers/UsersController.php	18
app/Http/Controllers/VideosController.php	19
Crear els testos	20
COMPROVACIÓ TESTS:	20
tests/Feature/AuthApiTest.php	22
tests/Feature/MultimediaApiTest.php	23
tests/Feature/UsersApiTest.php	24
tests/Feature/VideosApiTest.php	25
Frontend (Ionic en vue):	26
Crear l'aplicació ionic en vue	26
Navbar i Footer	26
Connectar el login i registre amb l'api	28
src/plugins/axios.js	28
src/router/index.js	29
src/stores/auth.js	30
src/views/AuthView.vue	32
Vista principal amb els arxius multimedia	33
src/views/HomeView.vue	33
src/views/ProfileView.vue	34
CRUD de multimedia mitjançant l'api	36



src/views/CreateView.vue	36
src/views/EditView.vue	37
src/views/DeleteView.vue	38
src/views>ShowView.vue	39
COMPROVACIONS FINALS	41
LOGIN:	41
REGISTER:	41
HOME:	42
HOME:	42
SHOW:	43
EDIT:	43
DELETE:	44
PERFIL:	44
Generar Apk Android i comprovar el funcionament	45

Instal·lació Inicial

Paquets a instalar:

```
alumnat@harvey:~/Code/M7-8-9$ npm install -g npm@11.1.0
(node:10429) ExperimentalWarning: CommonJS module /home/alumnat/.nvm/versions/no
de/v23.1.0/lib/node_modules/npm/node_modules/debug/src/node.js is loading ES Mod
ule /home/alumnat/.nvm/versions/node/v23.1.0/lib/node_modules/npm/node_modules/s
upports-color/index.js using require().
Support for loading ES Module in require() is an experimental feature and might
change at any time
```

```
alumnat@harvey:~/Code/M7-8-9/IonicVideosAppHarvey$ npm install axios
(node:12330) ExperimentalWarning: CommonJS module /home/alumnat/.nvm/versions/no
de/v23.1.0/lib/node_modules/npm/node_modules/debug/src/node.js is loading ES Mod
ule /home/alumnat/.nvm/versions/node/v23.1.0/lib/node_modules/npm/node_modules/s
upports-color/index.js using require().
Support for loading ES Module in require() is an experimental feature and might
change at any time
(Use `node --trace-warnings ...` to show where the warning was created)
```

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$ npm install axios vue-filepond filepond @ionic/pwa-elements
(node:15880) ExperimentalWarning: CommonJS module /home/alumnat/.nvm/versions/node/v23.1.0/lib/node_modules/npm/nod
e_modules/debug/src/node.js is loading ES Module /home/alumnat/.nvm/versions/node/v23.1.0/lib/node_modules/npm/node_
modules/supports-color/index.js using require().
Support for loading ES Module in require() is an experimental feature and might change at any time
(Use `node --trace-warnings ...` to show where the warning was created)
```

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ sudo apt-get install php-gd
[sudo] contrasenya per a alumnat:
S'està llegint la llista de paquets... Fet
S'està construint l'arbre de dependències... Fet
S'està llegint la informació de l'estat... Fet
El paquets següents s'han instal·lat automàticament i ja no són necessaris:
fonts-tuffy gir1.2-ayatanaappindicator3-0.1 gir1.2-goocanvas-2.0 libarchive-zip-p
libcpanel-json-xs-perl libcrypt-openssl-bignum-perl libcrypt-openssl-random-perl
libgoocanvas2-cairotypes-perl libgoocanvas2-perl libgtk3-imageview-perl libimage-
libmime charset-perl libnet-dbus-glib-perl libnet-oauth-perl libnumber-bytes-huma
```

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$ npm install filepond vue-filepond filepond-plugin-file-validate-type filepond-plugin-image-preview
added 2 packages, and audited 723 packages in 4s

130 packages are looking for funding
  run 'npm fund' for details

4 vulnerabilities (2 moderate, 2 critical)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run 'npm audit' for details.
```

Backend (Laravel - VideosApp):

[Crear el localStorage](#)

database/factories/MultimediaFactory.php

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan make:factory MultimediaFactory --model=Multimedia
[INFO] Factory [database/factories/MultimediaFactory.php] created successfully.

alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$
```

Al multimedia especifiquem els camps a definir i el tipus.

```
© MultimediaFactory.php ×

1 <?php
2
3 namespace Database\Factories;
4
5 use App\Models\Multimedia;
6 use App\Models\User;
7 use Illuminate\Database\Eloquent\Factories\Factory;
8
9 no usages
10 class MultimediaFactory extends Factory
11 {
12     protected $model = Multimedia::class;
13
14     public function definition(): array
15     {
16         return [
17             'title' => $this->faker->sentence,
18             'description' => $this->faker->paragraph,
19             'path' => 'multimedia/'. $this->faker->uuid.'.jpg',
20             'type' => $this->faker->randomElement(['image', 'video']),
21             'user_id' => User::factory()
22         ];
23     }
}
```

config/filesystems.php

Al filesystems procuro tenir les rutes adequades de públic i media establint permisos.

```
'public' => [
    'driver' => 'local',
    'root' => storage_path('app/public'),
    'url' => env('key: APP_URL').'/storage',
    'visibility' => 'public',
    'throw' => false,
    'permissions' => [
        'file' => [
            'public' => 0664,
            'private' => 0664,
        ],
        'dir' => [
            'public' => 0775,
            'private' => 0775,
        ],
    ],
],
] ,
```



```
'media' => [ // Nou disk específic per a multimèdia
    'driver' => 'local',
    'root' => storage_path('app/public/media'),
    'url' => env('key: APP_URL').'/storage/media',
    'visibility' => 'public',
    'throw' => false,
    'permissions' => [
        'file' => [
            'public' => 0664,
            'private' => 0664,
        ],
        'dir' => [
            'public' => 0775,
            'private' => 0775,
        ],
    ],
],
```

Amb el storage:link conectare les dos rutes via un enllaç simbòlic.

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan storage:link
INFO The [public/storage] link has been connected to [storage/app/public].
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$
```

config/cors.php

```
php cors.php ×
1 <?php
2
3 return [
4     'paths' => ['api/*', 'sanctum/csrf-cookie', 'storage/*'],
5     'allowed_methods' => ['*'],
6     'allowed_origins' => ['http://localhost:8100'],
7     'allowed_origins_patterns' => [],
8     'allowed_headers' => ['*'],
9     'exposed_headers' => [],
10    'max_age' => 0,
11    'supports_credentials' => true,
12];
```

Crear el controller ApiMultimediaController

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan make:controller Api/ApiMultimediaController --api
INFO Controller [app/Http/Controllers/Api/ApiMultimediaController.php] created successfully.

alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$
```

Al apimultimediacontroller estableix o les funcions que farà servir el CRUD (index, store, update, delete, show). Cada funció específica els camps de la taula que afecta i accedeix al usuari i procurar retornar una resposta adequada.

INDEX:

```
② ApiMultimediaController.php ×
1  <?php
2
3  namespace App\Http\Controllers\Api;
4
5  > use ...
11
12 2 usages
12  class ApiMultimediaController extends Controller
13  {
14      /** Mostra tots els arxius multimedia */
15      public function index(): JsonResponse
16      {
17          $media = Multimedia::with('user:id,name,profile_photo_url')
18              ->latest()
19              ->get()
20              ->map(function ($item) {
21                  return [
22                      'id' => $item->id,
23                      'title' => $item->title,
24                      'description' => $item->description,
25                      'path' => Storage::url($item->path),
26                      'type' => $item->type,
27                      'user_id' => $item->user_id,
28                      'created_at' => $item->created_at->toDateTimeString(),
29                      'user' => $item->user
30                  ];
31              });
32
33          return response()->json([
34              'success' => true,
35              'data' => $media
36          ]);
37      }
}
```

STORE:

```
public function store(Request $request): JsonResponse
{
    $validated = $request->validate([
        'file' => 'required|file|mimes:mp4,jpeg,png,jpg|max:20480',
        'title' => 'required|string|max:255',
        'description' => 'nullable|string'
    ]);

    $file = $request->file('file');
    $path = $file->store(['path' => 'multimedia', 'options' => 'public']);

    $media = Multimedia::create([
        'title' => $validated['title'],
        'description' => $validated['description'],
        'path' => $path,
        'type' => str_starts_with($file->getMimeType(), 'video') ? 'video' : 'image',
        'user_id' => Auth::id()
    ]);

    return response()->json([
        'success' => true,
        'data' => $media->load('user:id,name'),
        'url' => Storage::url($path)
    ], status: 201);
}
```

STORE:

```
/** Mostra un arxiu específic */
public function show($id): JsonResponse
{
    try {
        $media = Multimedia::with(['user:id,name,profile_photo_url'])
            ->findOrFail($id);

        return response()->json([
            'success' => true,
            'data' => [
                'id' => $media->id,
                'title' => $media->title,
                'description' => $media->description,
                'path' => Storage::url($media->path),
                'type' => $media->type,
                'user_id' => $media->user_id,
                'created_at' => $media->created_at->toDateTimeString(),
                'user' => $media->user
            ]
        ]);
    } catch (\Exception $e) {
        return response()->json([
            'success' => false,
            'message' => 'Media not found'
        ], ['status: 404']);
    }
}
```

UPDATE:

```
public function update(Request $request, Multimedia $multimedia): JsonResponse
{
    if ($multimedia->user_id !== Auth::id()) {
        return response()->json([
            'success' => false,
            'message' => 'No autoritzat'
        ], status: 403);
    }

    // Validar les dades
    $validated = $request->validate([
        'file' => 'sometimes|file|mimes:mp4,jpeg,png,jpg|max:20480',
        'title' => 'sometimes|string|max:255',
        'description' => 'nullable|string'
    ]);

    // Gestionar l'actualització del fitxer si s'ha proporcionat
    if ($request->hasFile( key: 'file')) {
        // Eliminar el fitxer antic
        Storage::disk( name: 'public')->delete($multimedia->path);

        // Pujar el nou fitxer
        $file = $request->file( key: 'file');
        $path = $file->store( path: 'multimedia', options: 'public');

        // Actualitzar el path i el tipus
        $multimedia->path = $path;
        $multimedia->type = str_starts_with($file->getMimeType(), 'video') ? 'video' : 'image';
    }

    // Actualitzar les altres dades
    $multimedia->title = $validated['title'] ?? $multimedia->title;
    $multimedia->description = $validated['description'] ?? $multimedia->description;
    $multimedia->save();

    return response()->json([
        'success' => true,
        'message' => 'Multimedia actualitzat',
        'data' => $multimedia->fresh(['user:id,name']),
        'url' => $multimedia->fresh()->full_url // Assegura't que tens l'atribut full_url al model
    ]);
}
```



DESTROY/DELETE:

```
public function destroy(Multimedia $multimedia): JsonResponse
{
    // Verifica que l'usuari està autenticat
    if (!auth()->check()) {
        return response()->json([
            'success' => false,
            'message' => 'No autenticat'
        ], status: 401);
    }

    // Verifica que l'usuari és el propietari
    if ($multimedia->user_id !== auth()->id()) {
        return response()->json([
            'success' => false,
            'message' => 'No tens permisos per eliminar aquest contingut'
        ], status: 403);
    }

    try {
        // Elimina el fitxer físic
        Storage::disk( name: 'public')->delete($multimedia->path);

        // Elimina el registre de la base de dades
        $multimedia->delete();

        return response()->json([
            'success' => true,
            'message' => 'Contingut eliminat correctament'
        ]);
    } catch (\Exception $e) {
        return response()->json([
            'success' => false,
            'message' => 'Error en eliminar el contingut',
            'error' => $e->getMessage()
        ], status: 500);
    }
}
```

Crear el Model

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan make:model Multimedia -m
INFO Model [app/Models/Multimedia.php] created successfully.

INFO Migration [database/migrations/2025_03_24_153958_create_multimedia_table.php] created successfully.

alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$
```

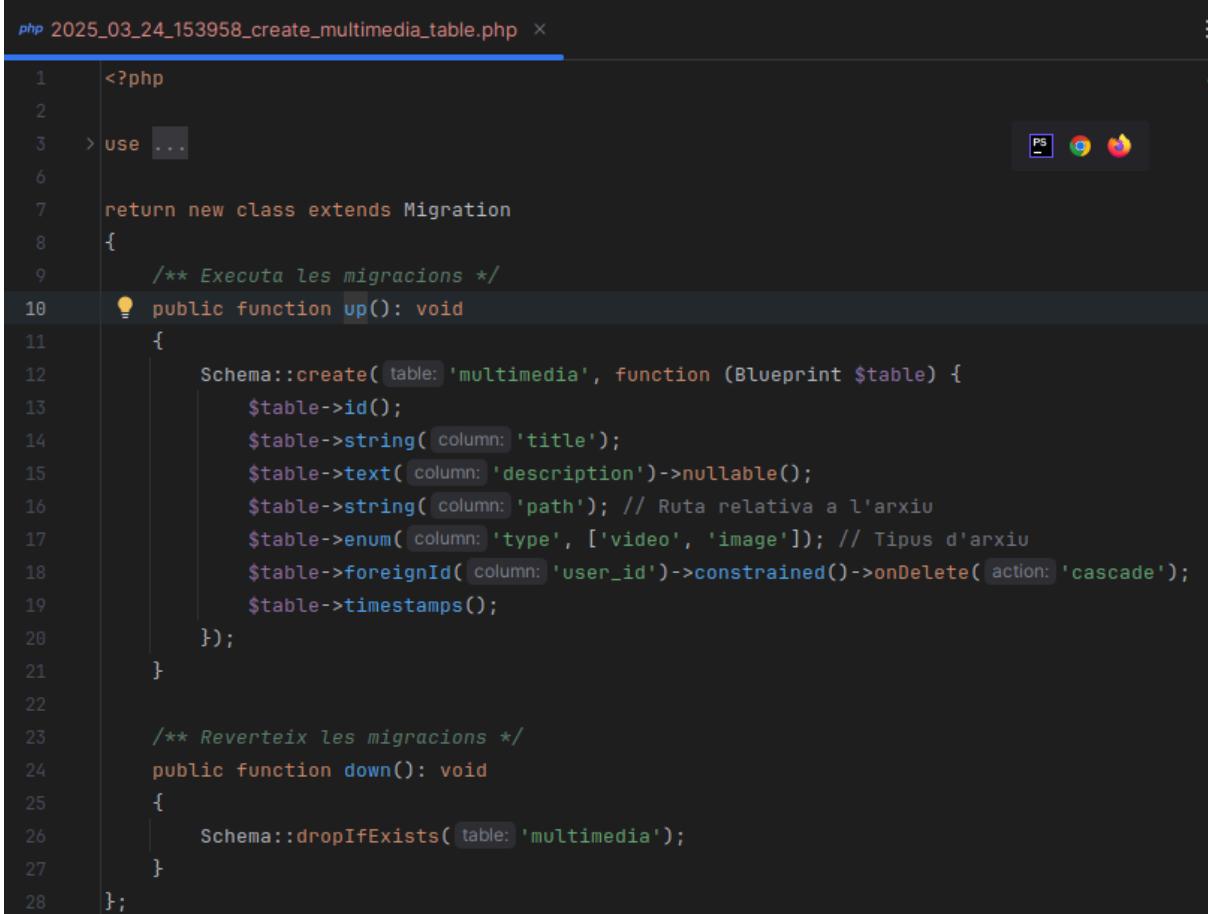
app/Models/Multimedia.php

Al fitxer de multimedia procuro tenir el fillable per especificar els camps, appens per incloure la url i 2 funcions per l'usuari i l'esborrat del multimedia.

```
⑤ Multimedia.php ×
  24
  25     14 usages
  26 class Multimedia extends Model
  27 {
  28     ⑥ protected $casts = [];
  29     ⑥ protected $fillable = ['id', 'title', 'description', 'path', 'type', 'user_id'];
  30     ⑥ protected $appends = ['full_url'];
  31
  32     no usages
  33     public function getFullUrlAttribute(): string
  34     {
  35         return Storage::disk('name:public')->url($this->path);
  36     }
  37
  38     public function user(): BelongsTo
  39     {
  40         return $this->belongsTo(related: User::class)->withDefault([
  41             'name' => 'Anonymous',
  42             'profile_photo_url' => asset(path: 'default-avatar.png')
  43         ]);
  44     }
  45
  46     ⑥ /**
  47     * Esborrat físic del fitxer en cascada */
  48     ⑥ protected static function booted(): void
  49     {
  50         static::deleting(function (Multimedia $multimedia) {
  51             if (Storage::disk('name:public')->exists($multimedia->path)) {
  52                 Storage::disk('name:public')->delete($multimedia->path);
  53             }
  54         });
  55     }
}
```

database/migrations/2025_03_24_153958_create_multimedia_table.php

La taula de creació de multimedia ha de tindre els següents camps i tipus:



The screenshot shows a code editor window with a dark theme. The file is named '2025_03_24_153958_create_multimedia_table.php'. The code defines a database migration class that creates a 'multimedia' table with the following columns:

- 'id' (integer, primary key)
- 'title' (string)
- 'description' (text, nullable)
- 'path' (string, relative path to the file)
- 'type' (enum, values: 'video', 'image')
- 'user_id' (foreign key referencing 'user_id' in another table, cascade delete)
- 'timestamps' (auto-generated)

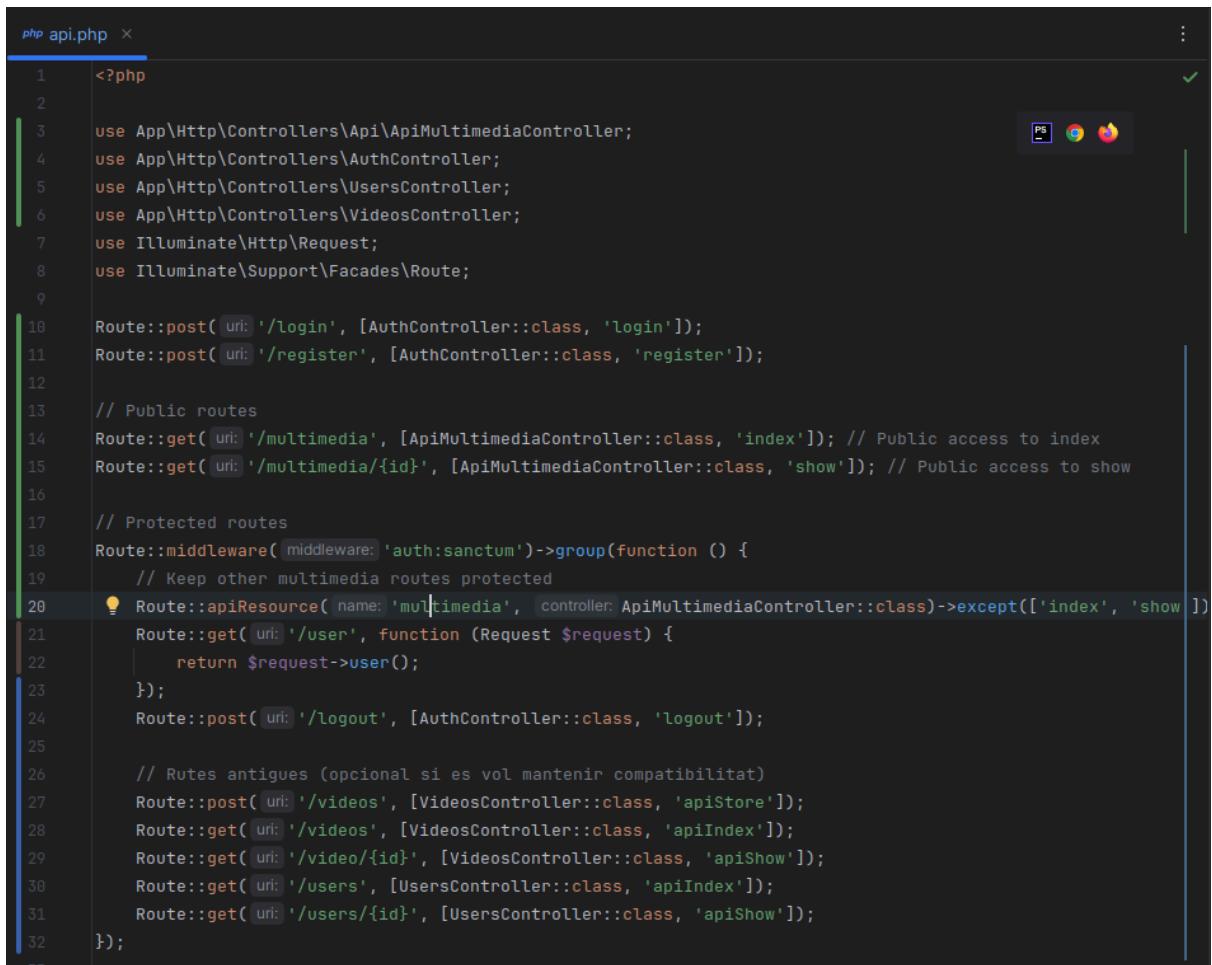
The code also includes a 'down()' method to drop the table if it exists.

```
1 <?php
2
3 > use ...
4
5
6
7 return new class extends Migration
8 {
9
10     /** Executa les migracions */
11     public function up(): void
12     {
13         Schema::create('multimedia', function (Blueprint $table) {
14             $table->id();
15             $table->string('title');
16             $table->text('description')->nullable();
17             $table->string('path'); // Ruta relativa a l'arxiu
18             $table->enum('type', ['video', 'image']); // Tipus d'arxiu
19             $table->foreignId('user_id')->constrained()->onDelete('cascade');
20             $table->timestamps();
21         });
22     }
23
24     /** Reverteix les migracions */
25     public function down(): void
26     {
27         Schema::dropIfExists('multimedia');
28     }
};
```

[Crear la api](#)

routes/api.php

Les rutes del api han de contenir les originals de login, register, vídeos i usuaris procurant d'especificar el store, show i index però de l'api. Com a adicional, ha de tenir l'usuari individual i el multimedia que funciona a través del ApiMultimediaController.



```
php api.php ×
1 <?php
2
3 use App\Http\Controllers\Api\ApiMultimediaController;
4 use App\Http\Controllers\AuthController;
5 use App\Http\Controllers\UsersController;
6 use App\Http\Controllers\VideosController;
7 use Illuminate\Http\Request;
8 use Illuminate\Support\Facades\Route;
9
10 Route::post('login', [AuthController::class, 'login']);
11 Route::post('register', [AuthController::class, 'register']);
12
13 // Public routes
14 Route::get('multimedia', [ApiMultimediaController::class, 'index']); // Public access to index
15 Route::get('multimedia/{id}', [ApiMultimediaController::class, 'show']); // Public access to show
16
17 // Protected routes
18 Route::middleware('auth:sanctum')->group(function () {
19     // Keep other multimedia routes protected
20     Route::apiResource('multimedia', ApiMultimediaController::class)->except(['index', 'show']);
21     Route::get('/user', function (Request $request) {
22         return $request->user();
23     });
24     Route::post('/logout', [AuthController::class, 'logout']);
25
26     // Rutes antigues (opcional si es vol mantenir compatibilitat)
27     Route::post('/videos', [VideosController::class, 'apiStore']);
28     Route::get('/videos', [VideosController::class, 'apiIndex']);
29     Route::get('/video/{id}', [VideosController::class, 'apiShow']);
30     Route::get('/users', [UsersController::class, 'apiIndex']);
31     Route::get('/users/{id}', [UsersController::class, 'apiShow']);
32 });

```

app/Http/Controllers/AuthController.php

Al Authcontroller el login, register i logout per a fer les funcions al frontend a través de l'api.

LOGIN:

```
AuthController.php ×
1 <?php
2
3 namespace App\Http\Controllers;
4
5 > use ...
11
12     4 usages
13 class AuthController extends Controller
14 {
15     /** Inici de sessió d'usuari */
16     public function login(Request $request): JsonResponse
17     {
18         $request->validate([
19             'email' => 'required|email',
20             'password' => 'required',
21             'device_name' => 'required',
22         ]);
23
24         $user = User::where('email', $request->email)->first();
25
26         if (! $user || ! Hash::check($request->password, $user->password)) {
27             throw ValidationException::withMessages([
28                 'email' => ['Les credencials proporcionades són incorrectes.'],
29             ]);
30
31         return response()->json([
32             'token' => $user->createToken($request->device_name)->plainTextToken,
33             'user' => $user
34         ]);
35     }
}
```

REGISTER:

```
/** Registre de nou usuari */
public function register(Request $request): JsonResponse
{
    $request->validate([
        'name' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:users',
        'password' => 'required|string|confirmed|min:8',
        'device_name' => 'required',
    ]);

    $user = User::create([
        'name' => $request->name,
        'email' => $request->email,
        'password' => Hash::make($request->password),
    ]);

    return response()->json([
        'token' => $user->createToken($request->device_name)->plainTextToken,
        'user' => $user
    ], [status: 201]);
}
```

LOGOUT:

```
/** Tancament de sessió */
1usage
public function logout(Request $request): JsonResponse
{
    $request->user()->currentAccessToken()->delete();

    return response()->json(['message' => 'Sessió tancada correctament']);
}
```

app/Http/Controllers/UsersController.php

Al UsersController he d'afegir les funcions com ara index i show però de l'api per a que funcionin.

```
② UsersController.php ×

12 class UsersController extends Controller
13 {
14     /**
15      * @usage new *
16      * @param Request $request
17      * @return JsonResponse
18      */
19
20     public function apiIndex(Request $request): JsonResponse
21     {
22         $query = User::query();
23
24         if ($search = $request->get('search')) {
25             $query->where('name', 'like', "%{$search}%")
26             ->orWhere('email', 'like', "%{$search}%");
27         }
28
29         $users = $query->get();
30
31         return response()->json([
32             'success' => true,
33             'data' => $users
34         ]);
35     }
36
37
38     /**
39      * Retorna un usuari específic per a l'API (nou mètode)
40      * @param int $id
41      * @return JsonResponse
42      */
43
44     /**
45      * @usage new *
46      * @param int $id
47      * @return JsonResponse
48      */
49
50     public function apiShow(int $id): JsonResponse
51     {
52         $user = User::with('videos')->findOrFail($id);
53
54         return response()->json([
55             'success' => true,
56             'data' => [
57                 'user' => $user,
58                 'videos' => $user->videos
59             ]
60         ]);
61     }
62 }
```

app/Http/Controllers/VideosController.php

Al VideosController he d'afegir les funcions com ara index i show però de l'api per a que funcionin.

```
© VideosController.php ×

13 class VideosController extends Controller
14
15     public function apiIndex(): JsonResponse
16     {
17
18         $videos = Video::whereNotNull('published_at')
19             ->with('user')
20             ->get();
21
22
23         return response()->json([
24             'success' => true,
25             'data' => $videos
26         ]);
27     }
28
29
30     /**
31      * Retorna un vídeo específic per a l'API (nou mètode)
32      * @param int $id
33      * @return JsonResponse
34      */
35
36     usage new *
37     public function apiShow(int $id): JsonResponse
38     {
39
40         $video = Video::with('user')->findOrFail($id);
41
42
43         if (!$video->published_at) {
44             return response()->json([
45                 'success' => false,
46                 'message' => 'Aquest vídeo no està disponible.'
47             ], 404);
48         }
49
50
51         return response()->json([
52             'success' => true,
53             'data' => $video
54         ]);
55     }
56
57 }
```

Crear els testos

Creo els 4 tests necessaris per comprovar els canvis.

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan make:test MultimediaApiTest
php artisan make:test AuthApiTest
php artisan make:test VideosApiTest
php artisan make:test UsersApiTest

[INFO] Test [tests/Feature/MultimediaApiTest.php] created successfully.

[INFO] Test [tests/Feature/AuthApiTest.php] created successfully.

[INFO] Test [tests/Feature/VideosApiTest.php] created successfully.

[INFO] Test [tests/Feature/UsersApiTest.php] created successfully.
```

COMPROVACIÓ TESTS:

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan test --filter MultimediaApiTest

[PASS] Tests\Feature\MultimediaApiTest
✓ pot obtenir llistat multimedia
✓ pot pujar nou arxiu multimedia
✓ pot eliminar arxiu multimedia

Tests:    3 passed (28 assertions)
Duration: 0.21s
```

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan test --filter AuthApiTest

[PASS] Tests\Feature\AuthApiTest
✓ usuari pot iniciar sessio
✓ credencials invalides retornen error
✓ usuari pot registrarse
✓ usuari pot tancar sessio

Tests:    4 passed (16 assertions)
Duration: 0.20s
```

```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan test --filter UsersApiTest

  PASS  Tests\Feature\UsersApiTest
    ✓ pot obtenir llistat usuaris
    ✓ pot fer cerques d usuaris
    ✓ pot obtenir usuari específic
    ✓ no autoritzat sense token

Tests:  4 passed (37 assertions)
Duration: 0.23s
```

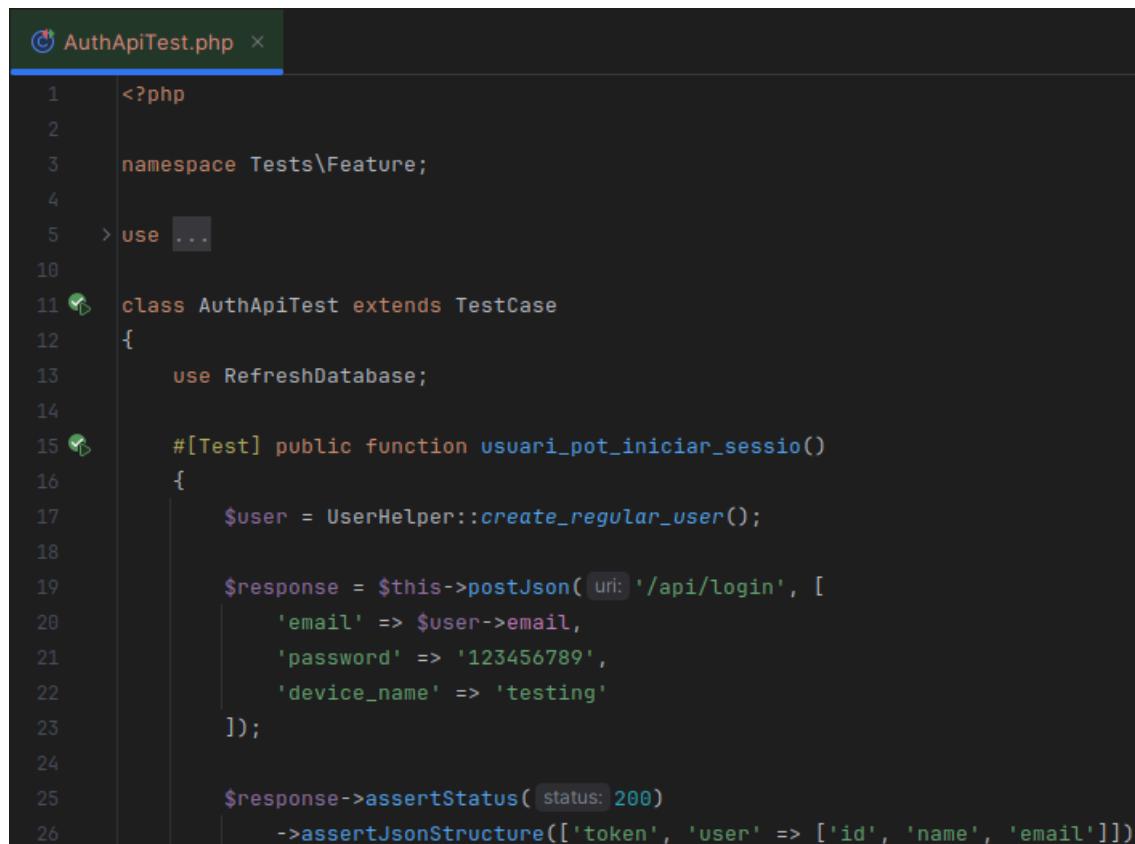
```
alumnat@harvey:~/Code/M7-8-9/VideoAppHarvey$ php artisan test --filter VideosApiTest

  PASS  Tests\Feature\VideosApiTest
    ✓ pot obtenir llistat videos
    ✓ pot obtenir video específic
    ✓ video no publicat retorna error
    ✓ pot crear video

Tests:  4 passed (11 assertions)
Duration: 0.20s
```

tests/Feature/AuthApiTest.php

Al AuthApiTest he comprovat la autenticació dels usuaris.

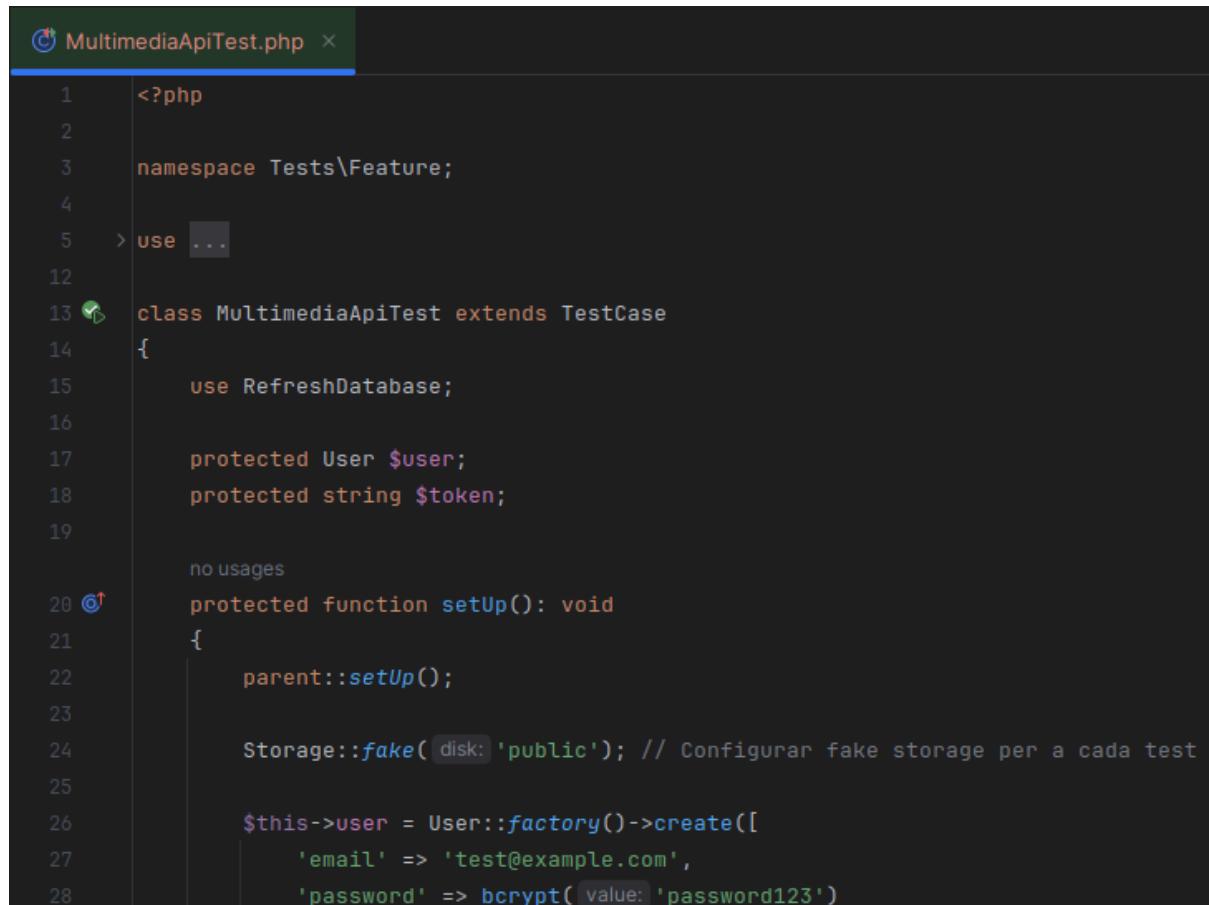


```
AuthApiTest.php ×
1  <?php
2
3  namespace Tests\Feature;
4
5  > use ...
10
11 class AuthApiTest extends TestCase
12 {
13     use RefreshDatabase;
14
15     #[Test] public function usuari_pot_iniciar_sessio()
16     {
17         $user = UserHelper::create_regular_user();
18
19         $response = $this->postJson( uri: '/api/login', [
20             'email' => $user->email,
21             'password' => '123456789',
22             'device_name' => 'testing'
23         ]);
24
25         $response->assertStatus( status: 200 )
26             ->assertJsonStructure([ 'token', 'user' => [ 'id', 'name', 'email' ] ])

```

tests/Feature/MultimediaApiTest.php

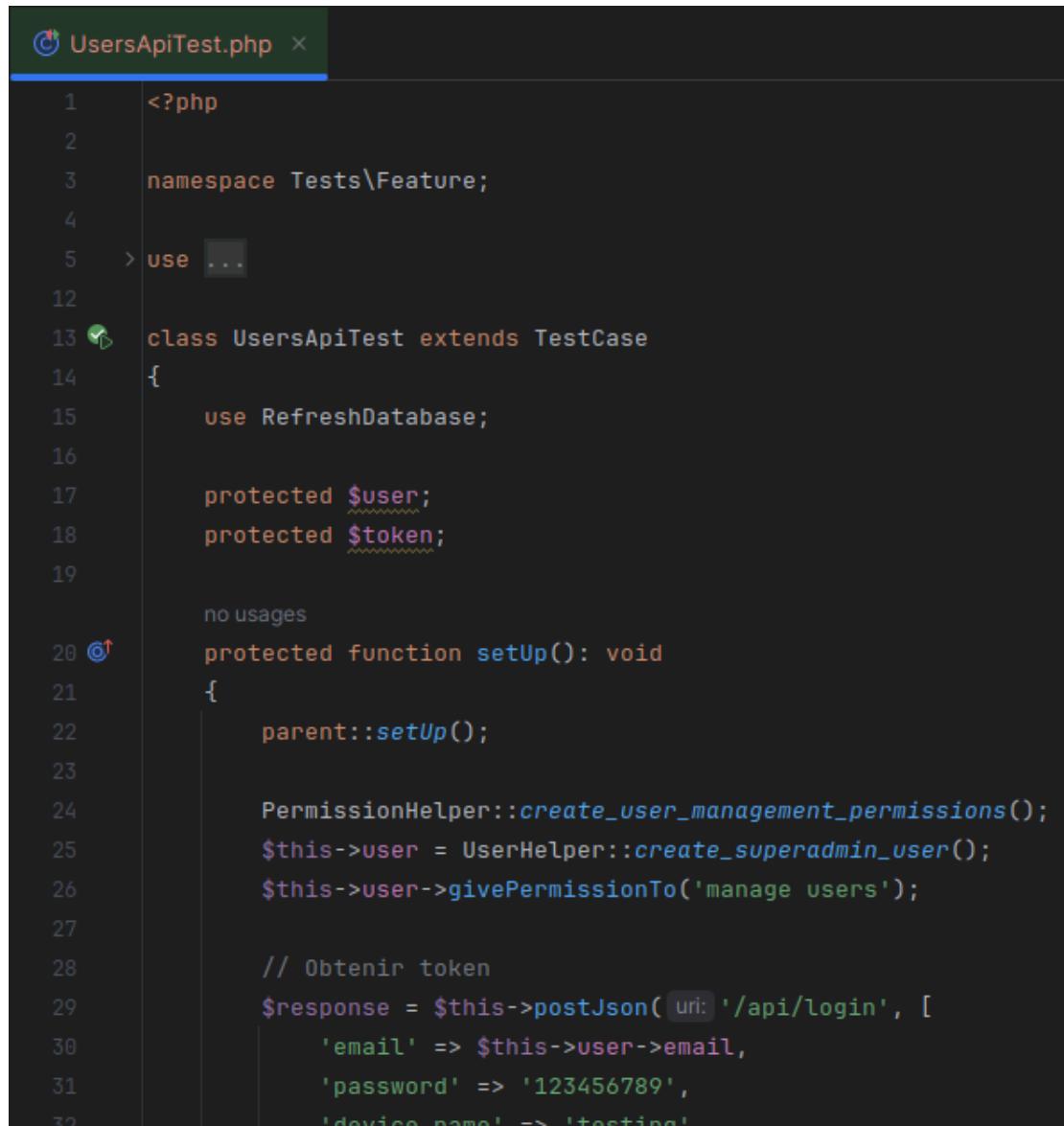
Al MultimediaApiTest he comprovat si la multimedia si es pot muntar, llistar etc...



```
1 <?php
2
3 namespace Tests\Feature;
4
5 > use ...
12
13 class MultimediaApiTest extends TestCase
14 {
15     use RefreshDatabase;
16
17     protected User $user;
18     protected string $token;
19
20     no usages
20 ⌂ protected function setUp(): void
21     {
22         parent::setUp();
23
24         Storage::fake( disk: 'public' ); // Configurar fake storage per a cada test
25
26         $this->user = User::factory()->create([
27             'email' => 'test@example.com',
28             'password' => bcrypt( value: 'password123' )
```

tests/Feature/UsersApiTest.php

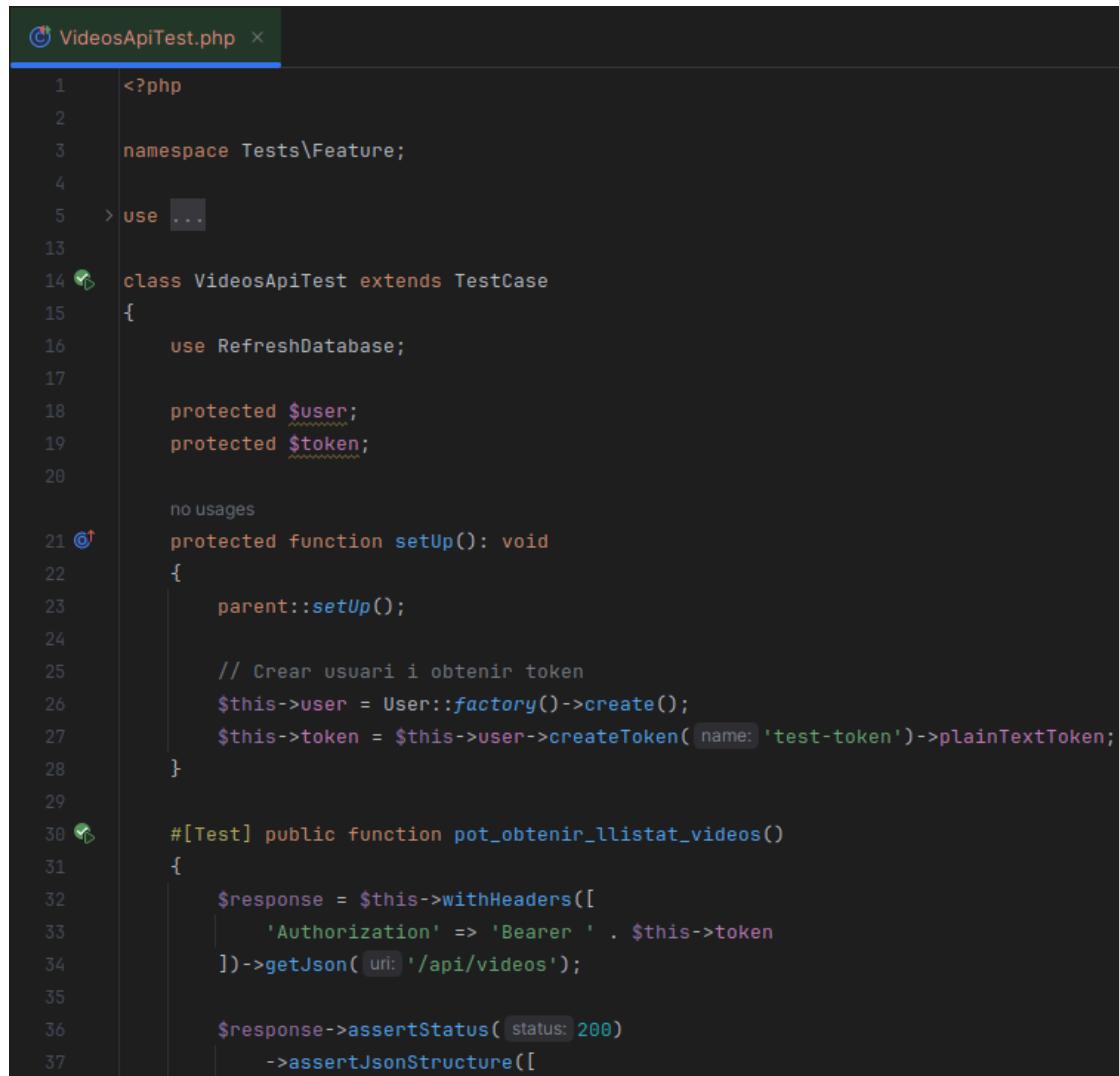
Al UsersApiTest he comprovat si els usuaris es poden crear, llistar etc...



```
1 <?php
2
3 namespace Tests\Feature;
4
5 > use ...
12
13 class UsersApiTest extends TestCase
14 {
15     use RefreshDatabase;
16
17     protected $user;
18     protected $token;
19
20     no usages
21     protected function setUp(): void
22     {
23         parent::setUp();
24
25         PermissionHelper::create_user_management_permissions();
26         $this->user = UserHelper::create_superuser();
27         $this->user->givePermissionTo('manage users');
28
29         // Obtenir token
30         $response = $this->postJson( uri: '/api/login', [
31             'email' => $this->user->email,
32             'password' => '123456789',
33             'device_name' => 'testing'
34         ]);
35
36         $this->token = $response->json['token'];
37     }
38
39     protected function tearDown(): void
40     {
41         $this->refreshDatabase();
42     }
43 }
```

tests/Feature/VideosApiTest.php

Al VideosApiTest he comprovat si els vídeos es poden crear, llistar etc...



The screenshot shows a code editor window with the file 'VideosApiTest.php' open. The code is a PHPUnit test for an API endpoint. It includes setup logic for creating a user and generating a token, followed by a test method that sends a GET request to the '/api/videos' endpoint and asserts the response status and JSON structure.

```
<?php

namespace Tests\Feature;

use ...;

class VideosApiTest extends TestCase
{
    use RefreshDatabase;

    protected $user;
    protected $token;

    no usages

    protected function setUp(): void
    {
        parent::setUp();

        // Crear usuari i obtenir token
        $this->user = User::factory()->create();
        $this->token = $this->user->createToken( name: 'test-token' )->plainTextToken;
    }

    #[Test] public function pot_obtenir_llistat_videos()
    {
        $response = $this->withHeaders([
            'Authorization' => 'Bearer ' . $this->token
        ])->getJson( uri: '/api/videos');

        $response->assertStatus( status: 200 )
            ->assertJsonStructure([

```

Frontend (Ionic en vue):

Crear l'aplicació ionic en vue

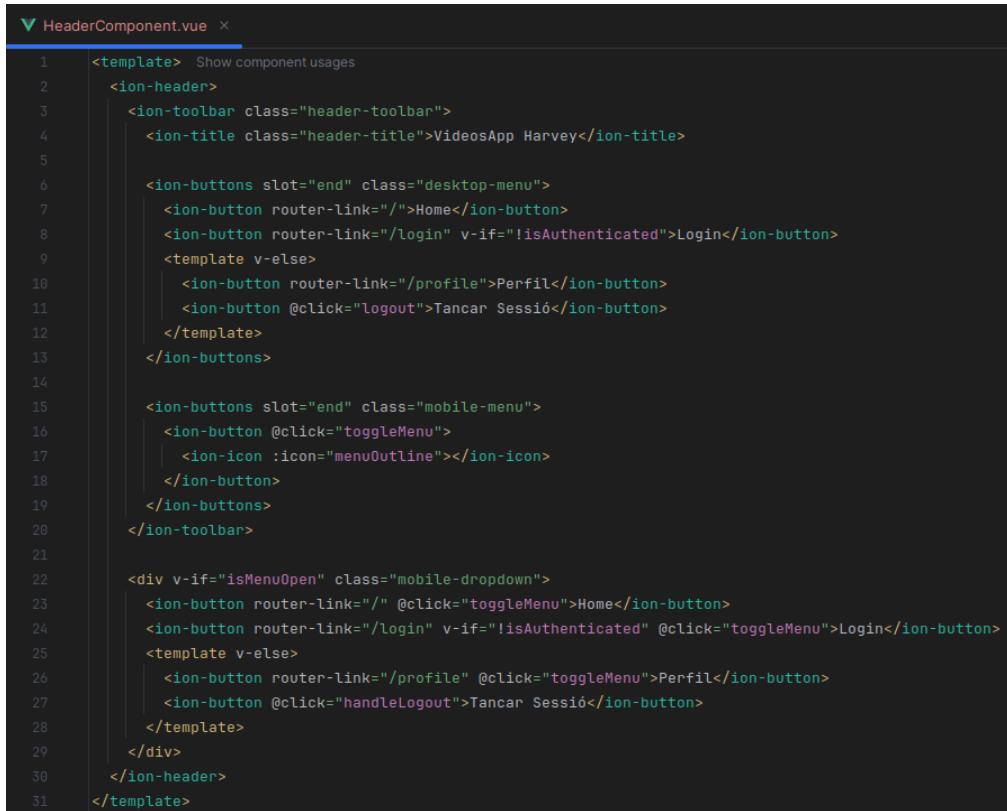
Creo l'app en blank d'ionic.

```
alumnat@harvey:~/Code/M7-8-9$ ionic start ionicVideosAppHarvey blank --type=vue
✓ Preparing directory ./ionicVideosAppHarvey in 743.89µs
✓ Downloading and extracting blank starter in 153.69ms
> ionic integrations enable capacitor --quiet -- ionicVideosAppHarvey
io.ionic.starter
> npm i --save -E @capacitor/core@latest
```

Navbar i Footer

Procuro tenir els links adequats, estils i funcionalitats al header.

VideosApp Harvey HOME PERFIL TANCAR SESSIÓ



```

1  <template> Show component usages
2    <ion-header>
3      <ion-toolbar class="header-toolbar">
4        <ion-title class="header-title">VideosApp Harvey</ion-title>
5
6        <ion-buttons slot="end" class="desktop-menu">
7          <ion-button router-link="/">Home</ion-button>
8          <ion-button router-link="/login" v-if="!isAuthenticated">Login</ion-button>
9          <template v-else>
10            <ion-button router-link="/profile">Perfil</ion-button>
11            <ion-button @click="logout">Tancar Sessió</ion-button>
12          </template>
13        </ion-buttons>
14
15        <ion-buttons slot="end" class="mobile-menu">
16          <ion-button @click="toggleMenu">
17            <ion-icon :icon="menuOutline"></ion-icon>
18          </ion-button>
19        </ion-buttons>
20      </ion-toolbar>
21
22      <div v-if="isMenuOpen" class="mobile-dropdown">
23        <ion-button router-link="/" @click="toggleMenu">Home</ion-button>
24        <ion-button router-link="/login" v-if="!isAuthenticated" @click="toggleMenu">Login</ion-button>
25        <template v-else>
26          <ion-button router-link="/profile" @click="toggleMenu">Perfil</ion-button>
27          <ion-button @click="handleLogout">Tancar Sessió</ion-button>
28        </template>
29      </div>
30    </ion-header>
31  </template>
```

Procuro tenir el meu nom i incluir la data actual així actualitza amb temps amb estils i funcions adequats.

© 2025 VideosApp - Desenvolupat per Harvey

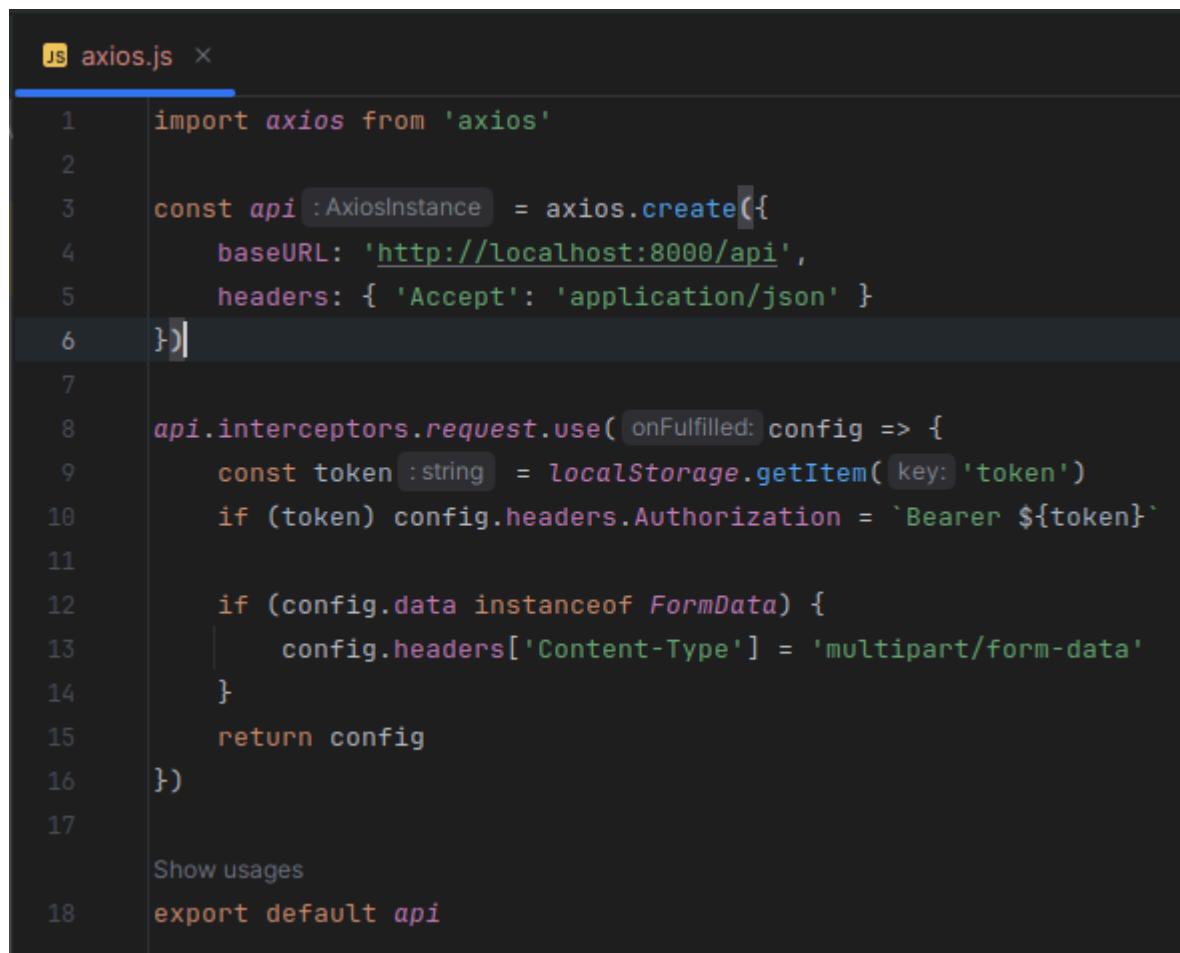
```
FooterComponent.vue  ×

1  <template> Show component usages
2    <ion-footer class="app-footer">
3      <ion-toolbar>
4        <div class="footer-content">
5          <div class="footer-copyright">
6            &copy; {{ new Date().getFullYear() }} VideosApp - Desenvolupat per Harvey
7          </div>
8        </div>
9      </ion-toolbar>
10     </ion-footer>
11   </template>
12
13 <script setup>
14 import { IonFooter, IonToolbar } from '@ionic/vue'
15 </script>
16
17 <style scoped>
18 .app-footer {
```

Connectar el login i registre amb l'api

src/plugins/axios.js

Crido la api amb el axios que guarda el token amb una variable reutilitzable per la resta del projecte.



```
JS axios.js ×

1 import axios from 'axios'
2
3 const api : AxiosInstance = axios.create({
4     baseURL: 'http://localhost:8000/api',
5     headers: { 'Accept': 'application/json' }
6 })
7
8 api.interceptors.request.use( onFulfilled: config => {
9     const token : string = localStorage.getItem( key: 'token' )
10    if (token) config.headers.Authorization = `Bearer ${token}`
11
12    if (config.data instanceof FormData) {
13        config.headers['Content-Type'] = 'multipart/form-data'
14    }
15    return config
16 })
17
18 Show usages
19 export default api
```

src/router/index.js

Al index del router estableix les rutes de l'app com ara el home, perfil, login, create, show, delete i edit.

```
js index.js ×
1 import { createRouter, createWebHistory } from '@ionic/vue-router'
2 import { useAuthStore } from '@/stores/auth'
3
4 const router : Router = createRouter( opts: {
5   history: createWebHistory(import.meta.env.BASE_URL),
6   routes: [
7     { path: '/', redirect: '/home' },
8     { path: '/home', component: () => import('@/views/HomeView.vue')},
9     { path: '/profile', component: () => import('@/views/ProfileView.vue'), meta: { requiresAuth: true }},
10    { path: '/login', component: () => import('@/views/AuthView.vue'), meta: { requiresGuest: true }},
11    { path: '/create', component: () => import('@/views/CreateView.vue'), meta: { requiresAuth: true }},
12    { path: '/multimedia/:id', component: () => import('@/views>ShowView.vue')},
13    { path: '/delete/:id', component: () => import('@/views>DeleteView.vue'), meta: { requiresAuth: true }},
14    { path: '/edit/:id', component: () => import('@/views>EditView.vue'), meta: { requiresAuth: true }}
15  ]
16})
17
```

Autentico cada pàgina que visita l'usuari per verificar qui és.

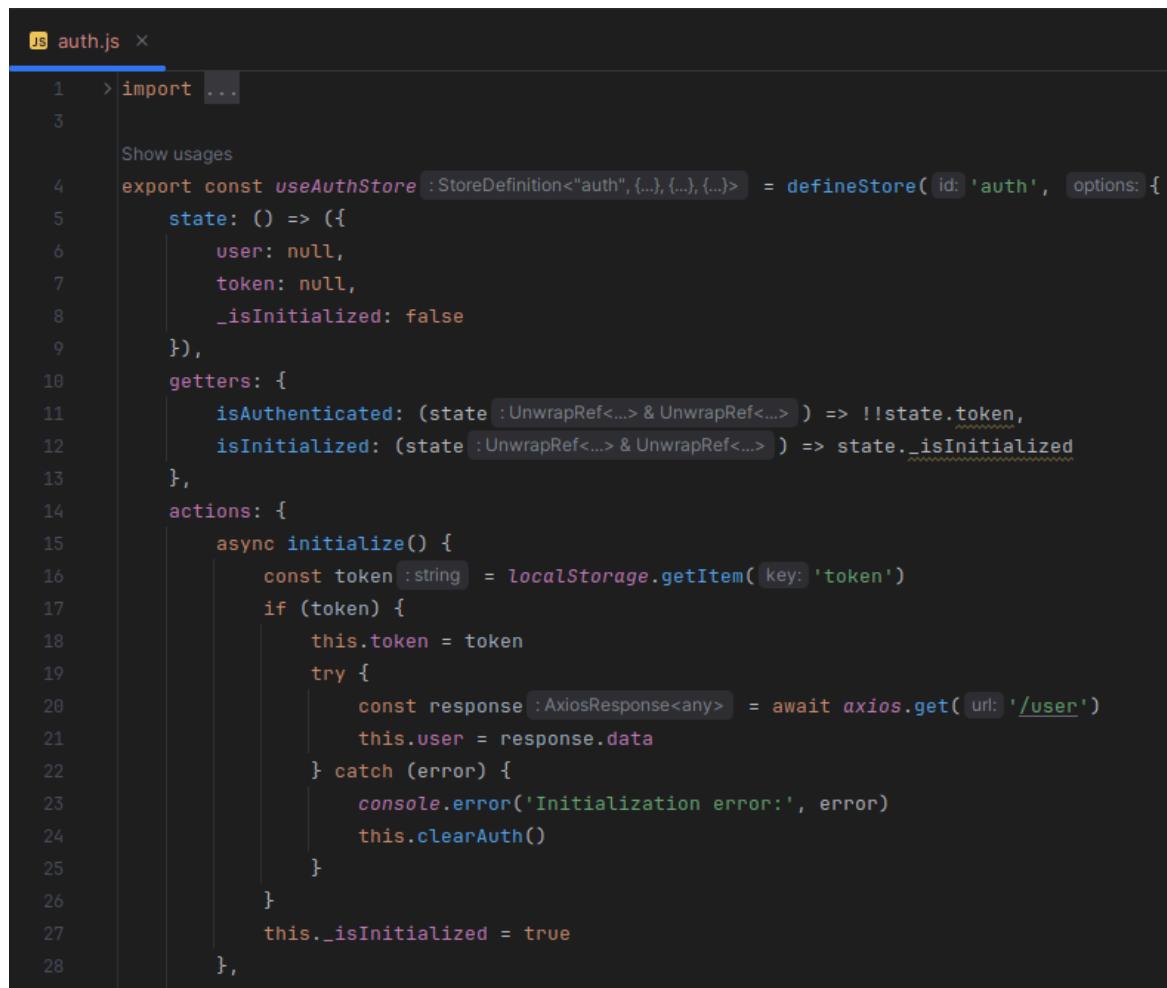
```
router.beforeEach( guard: async (to :RouteLocationNormalized , from :RouteLocationNormalizedLoaded , next :NavigationGuardNext ) => {
  const authStore :Store<"auth",{},{}> = useAuthStore()

  // Initialize the store if needed
  if (!authStore.isInitialized) {
    await authStore.initialize()
  }

  // Check if route requires authentication
  if (to.meta.requiresAuth && !authStore.isAuthenticated) {
    next('/login')
  }
  // Check if route requires guest (non-authenticated)
  else if (to.meta.requiresGuest && authStore.isAuthenticated) {
    next('/')
  }
  // Otherwise proceed
  else {
    next()
  }
})
```

src/stores/auth.js

Al auth verificar el token i cada pas del login,register i logout.



```
auth.js ×

1  > import ...
2
3
4  Show usages
5  export const useAuthStore : StoreDefinition<"auth", {}, {}> = defineStore( id: 'auth' , options: {
6      state: () => ({
7          user: null,
8          token: null,
9          _isInitialized: false
10     }),
11     getters: {
12         isAuthenticated: (state : UnwrapRef<...> & UnwrapRef<...>) => !!state.token,
13         isInitialized: (state : UnwrapRef<...> & UnwrapRef<...>) => state._isInitialized
14     },
15     actions: {
16         async initialize() {
17             const token : string = localStorage.getItem( key: 'token' )
18             if (token) {
19                 this.token = token
20                 try {
21                     const response : AxiosResponse<any> = await axios.get( url: '/user' )
22                     this.user = response.data
23                 } catch (error) {
24                     console.error('Initialization error:', error)
25                 }
26             }
27             this._isInitialized = true
28         },
29     }
30 }
```

Aquí veurem el login, register i logout amb cada pas que fan amb el token.

```
async login(credentials) {
    const formData = new FormData()
    formData.append('email', credentials.email)
    formData.append('password', credentials.password)
    formData.append('device_name', 'IonicVideosApp')

    try {
        const response : AxiosResponse<any> = await axios.post(url: '/login', formData)
        this.token = response.data.token
        this.user = response.data.user
        localStorage.setItem('token', this.token)
        return true
    } catch (error) {
        console.error('Login error:', error.response?.data)
        throw error // Propagar l'error per a maneig específic a la vista
    }
},

async register(userData) {
    const formData = new FormData()
    formData.append('name', userData.name)
    formData.append('email', userData.email)
    formData.append('password', userData.password)
    formData.append('password_confirmation', userData.password_confirmation)
    formData.append('device_name', 'IonicVideosApp')

    try {
        const response : AxiosResponse<any> = await axios.post(url: '/register', formData)
        this.token = response.data.token
        this.user = response.data.user
        localStorage.setItem('token', this.token)
        return true
    } catch (error) {
        console.error('Register error:', error.response?.data)
        throw error
    }
},

logout() {
    axios.post(url: '/logout')
    this.clearAuth()
},
```

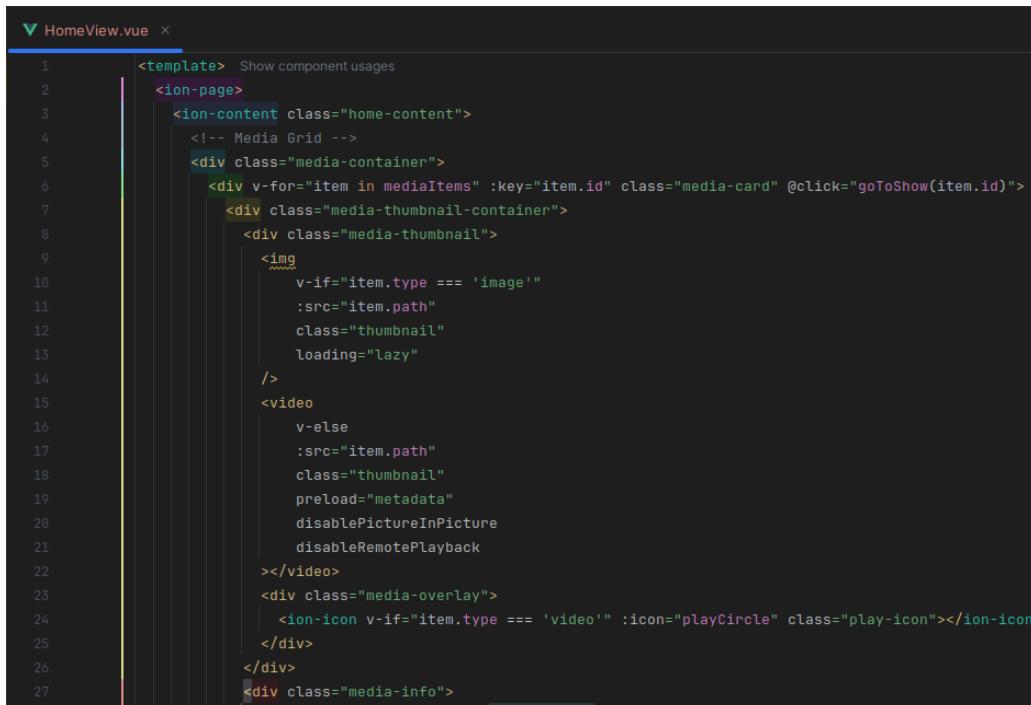
src/views/AuthView.vue

Al authview creo el login i register vinculat al backend amb una funció de handlesubmit per enviar la resposta. el !isLogin és per amagar els camps al login i mostrar altres al register.

```
V AuthView.vue X
1   <template> Show component usages
2     <ion-page>
3       <ion-content class="auth-content" scroll-y="false">
4         <div class="auth-wrapper">
5           <div class="auth-container">
6             <div class="auth-header">
7               <h1>{{ isLogin ? 'Iniciar Sessió' : 'Crear Compte' }}</h1>
8             </div>
9
10            <form @submit.prevent="handleSubmit" class="auth-form">
11              <ion-item v-if="!isLogin" class="form-item">
12                <ion-input
13                  v-model="form.name"
14                  type="text"
15                  :placeholder="!isLogin ? 'Nom' : ''"
16                  required
17                  :key="'name-' + isLogin"
18                ></ion-input>
19              </ion-item>
20
21              <ion-item class="form-item">
22                <ion-input
23                  v-model="form.email"
24                  type="email"
25                  placeholder="Email"
26                  required
27                  :key="'email-' + isLogin"
28                ></ion-input>
29              </ion-item>
30
31              <ion-item class="form-item">
32                <ion-input
33                  v-model="form.password"
34                  type="password"
35                  placeholder="Contrasenya"
36                  required
37                  :key="!isLogin ? 'password-' + isLogin" ></ion-input>
```

Vista principal amb els arxius multimedia

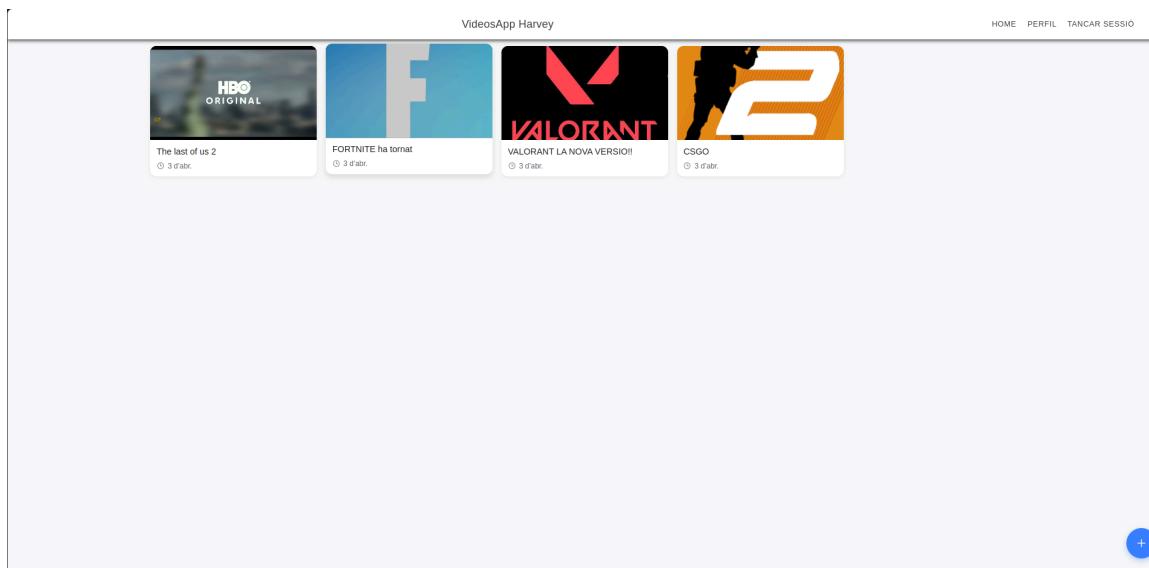
src/views/HomeView.vue



```

<template> Show component usages
<ion-page>
  <ion-content class="home-content">
    <!-- Media Grid -->
    <div class="media-container">
      <div v-for="item in mediaItems" :key="item.id" class="media-card" @click="goToShow(item.id)">
        <div class="media-thumbnail-container">
          <div class="media-thumbnail">
            
            <video
              v-else
              :src="item.path"
              class="thumbnail"
              preload="metadata"
              disablePictureInPicture
              disableRemotePlayback
            ></video>
          </div>
        </div>
        <div class="media-overlay">
          <ion-icon v-if="item.type === 'video'" :icon="playCircle" class="play-icon"></ion-icon>
        </div>
      </div>
      <div class="media-info">
    
```

Veurem la vista principal que té el contingut mostrat els vídeos amb el nom i data creat i un botó per creació de vídeos que sol apareix si estàs logeijat.



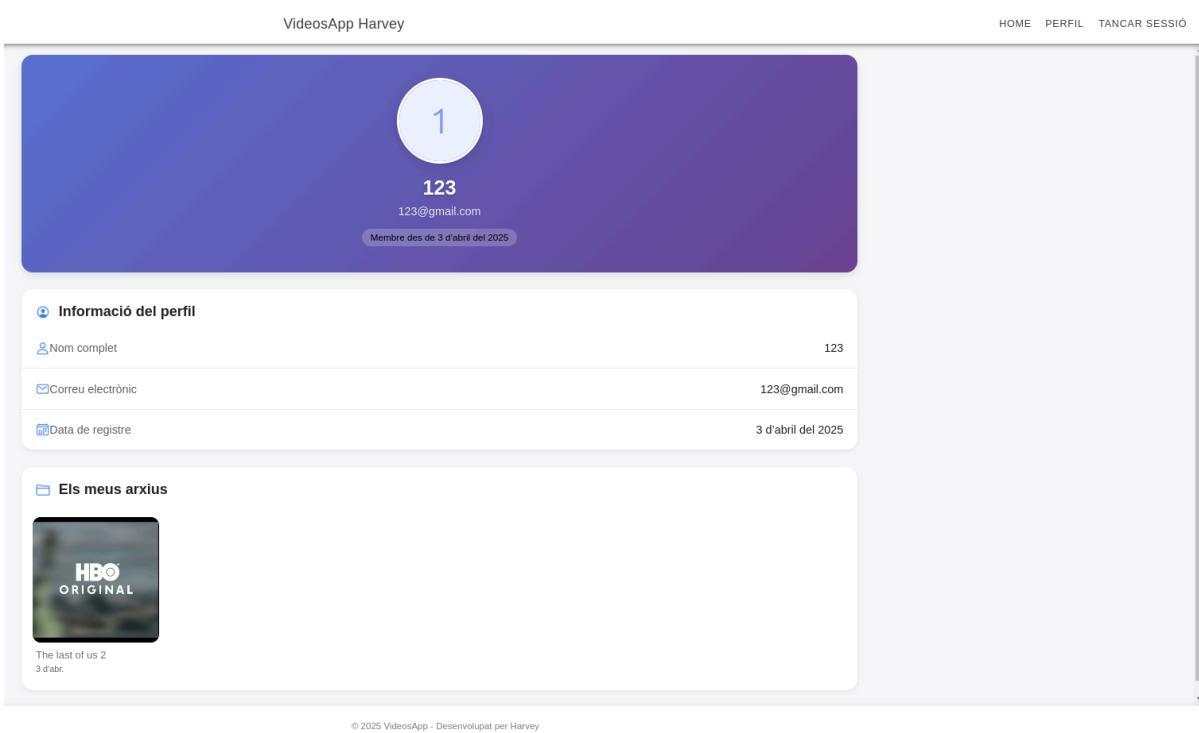
src/views/ProfileView.vue

Aquest component mostra el perfil d'usuari amb la seva informació personal i els arxius multimèdia pujats, amb un disseny visual atractiu i responsive. Inclou funcionalitats com visualització de dades, navegació a contingut i gestió d'estats de càrrega.

```
V ProfileView.vue X
1      <template> Show component usages
2      <ion-page>
3          <ion-content class="profile-content">
4              <div v-if="user" class="profile-container">
5                  <!-- Capçalera del perfil -->
6                  <div class="profile-header" :style="{ background: randomColor }">
7                      <ion-avatar class="profile-avatar">
8                          
9                      </ion-avatar>
10                     <div class="profile-info">
11                         <h1 class="profile-name">{{ user.name }}</h1>
12                         <p class="profile-email">{{ user.email }}</p>
13                         <ion-badge color="light" class="member-since">
14                             Membre des de {{ formatDate(user.created_at) }}
15                         </ion-badge>
16                     </div>
17                 </div>
18
19                 <!-- Informació del perfil -->
20                 <ion-card class="profile-card">
21                     <ion-card-header class="card-header">
22                         <ion-card-title class="card-title">
23                             <ion-icon :icon="personCircleOutline" class="title-icon"></ion-icon>
24                             Informació del perfil
25                         </ion-card-title>
26                     </ion-card-header>
27
28                     <ion-list lines="none" class="info-list">
29                         <ion-item class="info-item">
30                             <ion-icon :icon="personOutline" slot="start" class="info-icon"></ion-icon>
31                             <ion-label class="info-label">Nom complet</ion-label>
32                             <ion-text class="info-text">{{ user.name }}</ion-text>
33                         </ion-item>
34
35                         <ion-item class="info-item">
36                             <ion-icon :icon="mailOutline" slot="start" class="info-icon"></ion-icon>
37                             <ion-label class="info-label">Correu electrònic</ion-label>

```

Una vegadaloguejat, a la pàgina de l'usuari han d'aparèixer els seus arxius i ha de poder navegar a la pàgina principal on hi ha tots els arxius multimedia.



The screenshot shows a user profile page for 'VideosApp Harvey'. At the top, there's a navigation bar with links for 'HOME', 'PERFIL', and 'TANCAR SESSIÓ'. The main area features a large purple header with a circular profile picture containing the number '1', the name '123', and the email '123@gmail.com'. Below this, a message indicates the user has been a member since April 3, 2025. The page is divided into sections: 'Informació del perfil' (Profile information) which lists 'Nom complet' (123), 'Correu electrònic' (123@gmail.com), and 'Data de registre' (3 d'abril del 2025); and 'Els meus arxius' (My files) which displays a thumbnail for 'The last of us 2' (3 d'abr.) with the 'HBO ORIGINAL' logo. At the bottom, a footer note reads '© 2025 VideosApp - Desenvolupat per Harvey'.

CRUD de multimedia mitjançant l'api

S'ha d'utilitzar el Filepond per poder arrastrar els vídeos per a pujar-se al create.

src/views/CreateView.vue

Aquest component gestiona la pujada i visualització de contingut multimèdia, amb dues vistes principals: un formulari per pujar arxius (imatges/vídeos) amb títol i descripció, i una galeria que mostra tots els arxius pujats amb opcions d'edició i eliminació.

```
▼ CreateView.vue ×
1   <template> Show component usages
2     <ion-page>
3       <ion-content class="ion-padding">
4         <!-- Upload Form -->
5         <template v-if="$route.path === '/create'">
6           <form @submit.prevent="handleSubmit">
7             <div class="filepond-container">
8               <file-pond
9                 name="file"
10                ref="pond"
11                label-idle="Arrossegaa vídeo/imatge o <span class='filepond--label-action'>Buscar</span>"
12                accepted-file-types="image/jpeg, image/png, video/mp4"
13                max-file-size="20MB"
14                :server="serverConfig"
15                :instant-upload="false"
16                @processfile="handleFileSuccess"
17                @error="handleFileError"
18                @addfile="handleFileAdd"
19                @processfilestart="isUploading = true"
20                @processfileprogress="isUploading = true"
21              />
22            </div>
23
24            <ion-item class="ion-margin-top">
25              <ion-label position="stacked">Titol*</ion-label>
26              <ion-input v-model="form.title" required></ion-input>
27            </ion-item>
28
29            <ion-item class="ion-margin-top">
30              <ion-label position="stacked">Descripció</ion-label>
31              <ion-textarea v-model="form.description" rows="3"></ion-textarea>
32            </ion-item>
33
34            <ion-button
35              type="submit"
36              expand="block"
37              class="ion-margin-top"
38              :disabled="!isValid || isUploading">
39              <ion-spinner v-if="isUploading" name="crescent"></ion-spinner>
40              <span v-else>Publicar</span>
41            </ion-button>
42          </div>
```

src/views/EditView.vue

Component per editar publicacions multimèdia (títol, descripció i arxius) amb previsualització, validació de fitxers i gestió d'errors. Permet actualitzar contingut existent o pujar-ne de nou amb redirecció automàtica després dels canvis.

```
V EditView.vue X
1   <template> Show component usages
2     <ion-page>
3       <ion-content class="edit-content">
4         <form @submit.prevent="submitForm">
5           <ion-item>
6             <ion-input
7               label="Títol"
8               label-placement="stacked"
9               v-model="form.title"
10              required
11            ></ion-input>
12          </ion-item>
13
14          <ion-item>
15            <ion-textarea
16              label="Descripció"
17              label-placement="stacked"
18              v-model="form.description"
19              rows="4"
20              auto-grow
21            ></ion-textarea>
22          </ion-item>
23
24          <div class="current-media">
25            <h3>Contingut actual:</h3>
26            <video v-if="currentType === 'video'" :src="currentMediaPath" controls class="preview-media"></video>
27            
28          </div>
29
30          <div class="filepond-container">
31            <file-pond
32              name="file"
33              ref="pond"
34              label-idle="Arrossegua video/imatge o <span class='filepond--label-action'>Buscar</span>"
35              accepted-file-types="image/jpeg, image/png, video/mp4"
36              max-file-size="20MB"
37              :server="serverConfig"
38              :instant-upload="false"
39              @processfile="handleFileSuccess"
40              @error="handleFileError"
41              @addfile="handleFileAdd"
42              @processfilestart="isUploading = true"
43              @processfilecomplete="isUploading = false"
44            >
45          </div>
46        </ion-content>
47      </ion-page>
48    </template>
49
50    <script>
51      import { IonPage, IonContent, IonItem, IonInput, IonTextarea, IonText, IonVideo, FilePond } from '@ionic/vue';
52      import { ref, reactive } from 'vue';
53      import { useIonToast } from '@ionic/vue';
54      import { useIonRouter } from '@ionic/vue';
55      import { useIonLoading } from '@ionic/vue';
56      import { useIonActionSheet } from '@ionic/vue';
57      import { useIonModal } from '@ionic/vue';
58      import { useIonPopover } from '@ionic/vue';
59      import { useIonSegment } from '@ionic/vue';
60      import { useIonList } from '@ionic/vue';
61      import { useIonCard } from '@ionic/vue';
62      import { useIonGrid } from '@ionic/vue';
63      import { useIonRow } from '@ionic/vue';
64      import { useIonCol } from '@ionic/vue';
65      import { useIonFab, useIonFabButton, useIonFabClose } from '@ionic/vue';
66      import { useIonFabTray } from '@ionic/vue';
67      import { useIonFabPanel } from '@ionic/vue';
68      import { useIonFabCloseIcon } from '@ionic/vue';
69      import { useIonFabPanelIcon } from '@ionic/vue';
70      import { useIonFabCloseIcon } from '@ionic/vue';
71      import { useIonFabPanelIcon } from '@ionic/vue';
72      import { useIonFabCloseIcon } from '@ionic/vue';
73      import { useIonFabPanelIcon } from '@ionic/vue';
74      import { useIonFabCloseIcon } from '@ionic/vue';
75      import { useIonFabPanelIcon } from '@ionic/vue';
76      import { useIonFabCloseIcon } from '@ionic/vue';
77      import { useIonFabPanelIcon } from '@ionic/vue';
78      import { useIonFabCloseIcon } from '@ionic/vue';
79      import { useIonFabPanelIcon } from '@ionic/vue';
80      import { useIonFabCloseIcon } from '@ionic/vue';
81      import { useIonFabPanelIcon } from '@ionic/vue';
82      import { useIonFabCloseIcon } from '@ionic/vue';
83      import { useIonFabPanelIcon } from '@ionic/vue';
84      import { useIonFabCloseIcon } from '@ionic/vue';
85      import { useIonFabPanelIcon } from '@ionic/vue';
86      import { useIonFabCloseIcon } from '@ionic/vue';
87      import { useIonFabPanelIcon } from '@ionic/vue';
88      import { useIonFabCloseIcon } from '@ionic/vue';
89      import { useIonFabPanelIcon } from '@ionic/vue';
90      import { useIonFabCloseIcon } from '@ionic/vue';
91      import { useIonFabPanelIcon } from '@ionic/vue';
92      import { useIonFabCloseIcon } from '@ionic/vue';
93      import { useIonFabPanelIcon } from '@ionic/vue';
94      import { useIonFabCloseIcon } from '@ionic/vue';
95      import { useIonFabPanelIcon } from '@ionic/vue';
96      import { useIonFabCloseIcon } from '@ionic/vue';
97      import { useIonFabPanelIcon } from '@ionic/vue';
98      import { useIonFabCloseIcon } from '@ionic/vue';
99      import { useIonFabPanelIcon } from '@ionic/vue';
100     </script>
101
102     <style>
103       .current-media {
104         border: 1px solid #ccc;
105         padding: 10px;
106         margin-bottom: 10px;
107       }
108
109       .filepond-container {
110         border: 1px solid #ccc;
111         padding: 10px;
112       }
113     </style>
114   
```

src/views/DeleteView.vue

Component de confirmació per eliminar contingut multimèdia amb missatge d'avertència i botons d'accio/cancel·lar. Gestiona l'eliminació via API i redirigeix a l'inici amb actualització de dades.

```
V DeleteView.vue ×
1   <template> Show component usages
2     <ion-page>
3       <ion-content class="ion-padding">
4         <div class="confirmation-container">
5           <!-- Icôna d'avertència -->
6           <ion-icon :icon="warningOutline" color="warning" class="warning-icon"></ion-icon>
7           <h2>Confirmar eliminació</h2>
8           <p>Estàs segur que vols eliminar aquest contingut? Aquesta acció no es pot desfer.</p>
9
10          <div class="action-buttons">
11            <ion-button expand="block" fill="outline" @click="cancelDelete">Cancel·lar</ion-button>
12            <ion-button expand="block" color="danger" @click="deleteMedia" :disabled="loading">
13              <ion-spinner v-if="loading" name="crescent"></ion-spinner>
14              <span v-else>Eliminar</span>
15            </ion-button>
16          </div>
17        </div>
18      </ion-content>
19    </ion-page>
20  </template>
21
22  <script setup>
23  import { ref } from 'vue'
24  import { useRoute, useRouter } from 'vue-router'
25  import { warningOutline } from 'ionicons/icons'
26  import axios from '@/plugins/axios'
27  import {
28    IonPage,
29    IonContent,
30    IonButton,
31    IonIcon,
32    IonSpinner,
33    toastController
34  } from '@ionic/vue'
35
36  const route = useRoute()
37  const router = useRouter()
38  const loading = ref( value: false )
39
40  Show usages
41
```

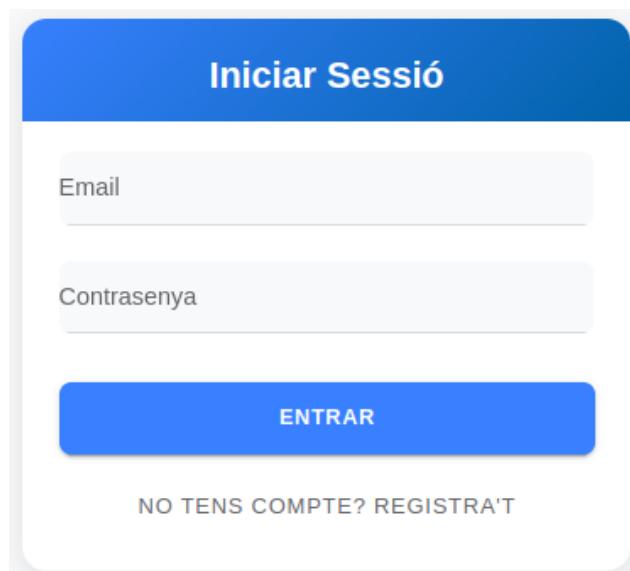
src/views>ShowView.vue

Pàgina de visualització de contingut multimèdia amb reproductor, informació detallada i llista de continguts relacionats. Inclou funcionalitats d'edició/eliminació per a propietaris i disseny responsive adaptat a tots els dispositius.

```
V ShowView.vue X
1  <template> Show component usages
2    <ion-page>
3      <ion-content class="media-detail-content">
4        <div class="media-layout">
5          <!-- Secció principal (esquerra) -->
6          <div class="main-content">
7            <!-- Reproductor de video/imatge -->
8            <div class="media-container">
9              <video
10                v-if="currentMedia.type === 'video'"
11                ref="videoPlayer"
12                :src="currentMedia.path"
13                controls
14                autoplay
15                class="media-player"
16                playsinline
17                disablePictureInPicture
18              ></video>
19              
24            </div>
25
26            <!-- Informació del contingut -->
27            <div class="media-info">
28              <h1 class="media-title">{{ currentMedia.title }}</h1>
29
30              <div class="media-meta">
31                <div class="author-info">
32                  <ion-avatar class="author-avatar">
33                    
36                    <span class="author-name">{{ currentMedia.user?.name || 'Anònim' }}</span>
37                    <span class="upload-date">{{ formatDate(currentMedia.created_at) }}</span>
38                  </div>
39                </div>
40
41                <div class="media-stats">
42                  <span class="stat-item">
43                    <span>{{ currentMedia.likes }}</span>
44                    <span>{{ currentMedia.comments }}</span>
45                  </span>
46                </div>
47              </div>
48            </div>
49
50            <div class="media-related">
51              <ion-grid>
52                <ion-row>
53                  <ion-col>
54                    <ion-card>
55                      <ion-card-header>
56                        <ion-card-subtitle>Contingut relatiu</ion-card-subtitle>
57                      <ion-card-content>
58                        <ion-list>
59                          <ion-item>Item 1</ion-item>
60                          <ion-item>Item 2</ion-item>
61                          <ion-item>Item 3</ion-item>
62                        </ion-list>
63                      </ion-card-content>
64                    </ion-card>
65                  </ion-col>
66                </ion-row>
67              </ion-grid>
68            </div>
69
70            <div class="media-actions">
71              <ion-grid>
72                <ion-row>
73                  <ion-col>
74                    <ion-button>Edicció</ion-button>
75                    <ion-button>Eliminar</ion-button>
76                  </ion-col>
77                </ion-row>
78              </ion-grid>
79            </div>
80
81          </div>
82        </ion-content>
83      </ion-page>
84    </div>
```

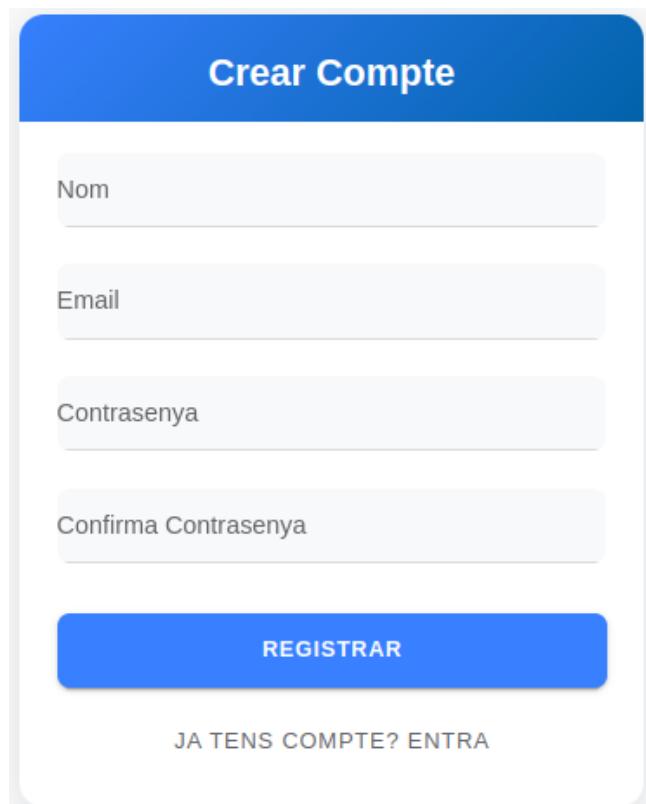
COMPROVACIONS FINALS

LOGIN:



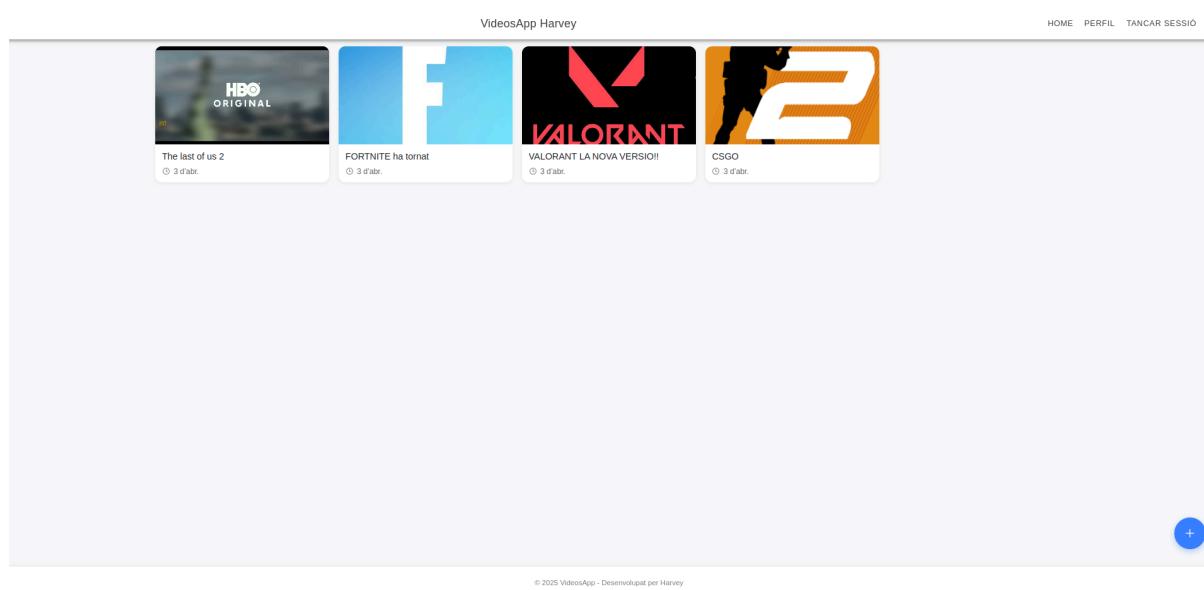
The login form features a blue header bar with the text "Iniciar Sessió". Below it are two input fields: "Email" and "Contrasenya", each with its own placeholder text. A large blue button labeled "ENTRAR" is centered below the inputs. At the bottom of the form, there is a link "NO TENS COMpte? REGISTRA'T".

REGISTER:

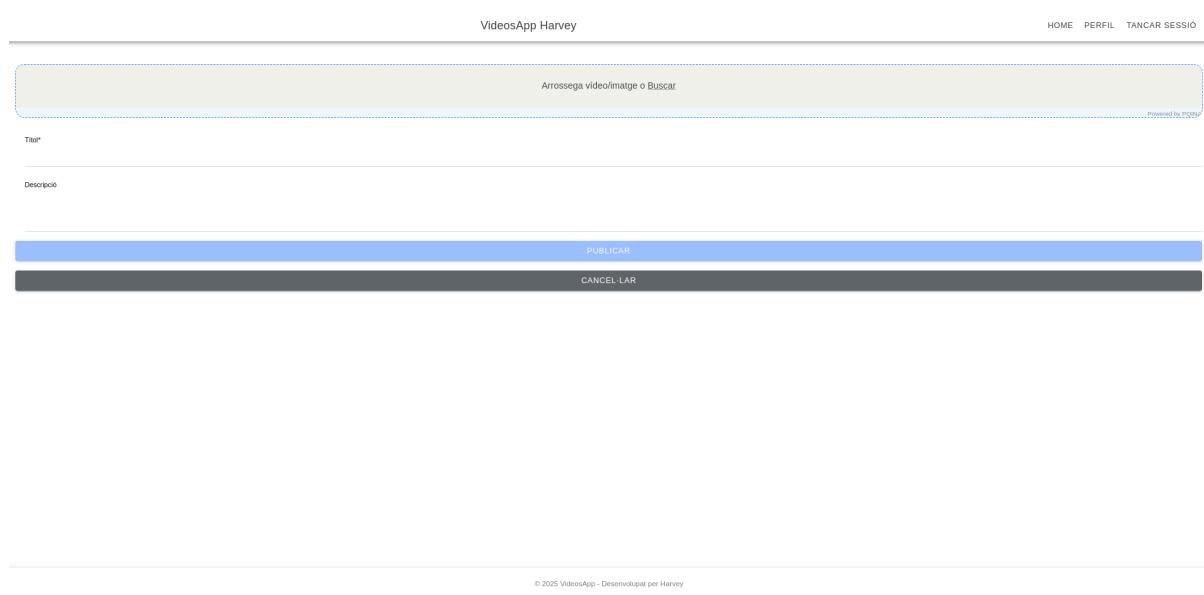


The register form features a blue header bar with the text "Crear Compte". Below it are four input fields: "Nom", "Email", "Contrasenya", and "Confirma Contrasenya", each with its own placeholder text. A large blue button labeled "REGISTRAR" is centered below the inputs. At the bottom of the form, there is a link "JA TENS COMpte? ENTRA".

HOME:



HOME:



SHOW:

VideosApp Harvey

HOME PERFIL TANCAR SESSIÓ

Més continguts

- FORTNITE ha tornat
Super Admin
8.613 vistes
- VALORANT LA NOVA VERSIÓ!
Super Admin
4.346 vistes
- CSGO
Super Admin
7.867 vistes

The last of us 2

1 123 3 d'abril del 2025 696.232 visualitzacions

bona serie limitada

EDITAR ELIMINAR

© 2025 VideosApp - Desenvolupat per Harvey

EDIT:

VideosApp Harvey

HOME PERFIL TANCAR SESSIÓ

Títol
The last of us 2

Descripció
bona serie limitada

Contingut actual:

Arrastra video/imatge o Buscar

PUBLICAR CANCEL·LAR

© 2025 VideosApp - Desenvolupat per Harvey



DELETE:

VideosApp Harvey

HOME PERFIL TANCAR SESSIÓ

⚠️

Confirmar eliminació

Estàs segur que vols eliminar aquest contingut? Aquesta acció no es pot desfer.

CANCEL·LAR

ELIMINAR

© 2025 VideosApp - Desenvolupat per Harvey

PERFIL:

VideosApp Harvey

HOME PERFIL TANCAR SESSIÓ

1

123

123@gmail.com

Membre des de 3 d'abril del 2025

Informació del perfil

Nom complet: 123

Correu electrònic: 123@gmail.com

Data de registre: 3 d'abril del 2025

Els meus arxius

The last of us 2
3 d'abril

© 2025 VideosApp - Desenvolupat per Harvey

Generar Apk Android i comprovar el funcionament

Comanda generant:

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$ ionic capacitor add android
> npm i -E @capacitor/android@7.1.0

added 2 packages, and audited 728 packages in 2s

130 packages are looking for funding
  run `npm fund` for details

4 vulnerabilities (2 moderate, 2 critical)

To address all issues (including breaking changes), run:
  npm audit fix --force
```

```
> capacitor add android
[capacitor] ✓ Adding native android project in android in 30.72ms
[capacitor] ✓ add in 31.40ms
[capacitor] ✓ Copying web assets from dist to android/app/src/main/assets/public in 52.91ms
[capacitor] ✓ Creating capacitor.config.json in android/app/src/main/assets in 545.85µs
[capacitor] ✓ copy android in 68.79ms
[capacitor] ✓ Updating Android plugins in 4.18ms
[capacitor] [info] Found 4 Capacitor plugins for android:
[capacitor]   @capacitor/app@7.0.0
[capacitor]   @capacitor/haptics@7.0.0
[capacitor]   @capacitor/keyboard@7.0.0
[capacitor]   @capacitor/status-bar@7.0.0
[capacitor] ✓ update android in 33.95ms
[capacitor] ✓ Syncing Gradle in 158.60µs
[capacitor] [success] android platform added!
[capacitor] Follow the Developer Workflow guide to get building:
[capacitor] https://capacitorjs.com/docs/basics/workflow
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$
```

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$ ionic build
> vite build
vite v5.2.14 building for production...
✓ 286 modules transformed.

dist/assets/status-tap-legacy-CvGi0hw7.js      0.60 kB | gzip:   0.40 kB
dist/assets/swipe-back-legacy-CqiL36U_.js        0.78 kB | gzip:   0.52 kB
dist/assets/focus-visible-legacy-Cd05cX4I.js     1.09 kB | gzip:   0.55 kB
dist/assets/md.transition-legacy-Jk8Cx0TZ.js    1.13 kB | gzip:   0.62 kB
dist/assets/keyboard2-legacy-Dglh4q-F.js         1.41 kB | gzip:   0.71 kB
dist/assets/index9-legacy-B9w2_hb1.js           1.74 kB | gzip:   0.87 kB
dist/assets/DeleteView-legacy-C0xuzVhY.js        2.43 kB | gzip:   1.28 kB
dist/assets/EditView-legacy-C40jJb3p.js          4.38 kB | gzip:   2.07 kB
dist/assets/AuthView-legacy-Cc0XGMVN.js          4.62 kB | gzip:   1.76 kB
dist/assets/input-shims-legacy-BV0EpVq2.js        4.74 kB | gzip:   2.07 kB
dist/assets/HomeView-legacy-a2KY0nQD.js          4.96 kB | gzip:   2.01 kB
dist/assets/CreateView-legacy-B5VuUBeL.js         7.22 kB | gzip:   3.12 kB
dist/assets>ShowView-legacy-CbPWos6D.js          7.90 kB | gzip:   2.90 kB
dist/assets/ProfileView-legacy-BZfMshlX.js       9.74 kB | gzip:   3.23 kB
dist/assets/ios.transition-legacy-B4z03qno.js   10.64 kB | gzip:   3.04 kB
dist/assets/polyfills-legacy-DYk20uP4.js          66.17 kB | gzip:  26.58 kB
dist/assets/filepond.min-legacy-CKBZav26.js     124.33 kB | gzip:  37.00 kB
dist/assets/index-legacy-CpJeCVTS.js            547.84 kB | gzip: 143.64 kB
```

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$ npx cap sync android
✓ Copying web assets from dist to android/app/src/main/assets/public in 53.08ms
✓ Creating capacitor.config.json in android/app/src/main/assets in 2.18ms
✓ copy android in 78.52ms
✓ Updating Android plugins in 4.11ms
[info] Found 4 Capacitor plugins for android:
  @capacitor/app@7.0.0
  @capacitor/haptics@7.0.0
  @capacitor/keyboard@7.0.0
  @capacitor/status-bar@7.0.0
✓ update android in 36.67ms
[info] Sync finished in 0.135s

Thank you for helping improve Capacitor by sharing anonymous usage data! ❤️
Information about the data we collect is available on our website: https://capacitorjs.com/docs/next/cli/telemetry
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey$
```

Executo el gradle per muntar el apk.

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey/android$ ./gradlew assembleDebug

> Configure project :app
WARNING: Using flatDir should be avoided because it doesn't support any meta-data formats.

> Configure project :capacitor-cordova-android-plugins
WARNING: Using flatDir should be avoided because it doesn't support any meta-data formats.

> Task :capacitor-android:compileDebugJavaWithJavac
Note: Some input files use unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

> Task :capacitor-status-bar:compileDebugJavaWithJavac
Note: /home/alumnat/Code/M7-8-9/ionicVideosAppHarvey/node_modules/@capacitor/status-bar/android/src/main/java/com/capacitorjs/plugins/statusbar/StatusBar.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
[Incubating] Problems report is available at: file:///home/alumnat/Code/M7-8-9/ionicVideosAppHarvey/android/build/reports/problems/problems-report.html
```

Aquí veurem el apk.

```
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey/android$ ls app/build/outputs/apk/debug/
app-debug.apk  output-metadata.json
alumnat@harvey:~/Code/M7-8-9/ionicVideosAppHarvey/android$
```