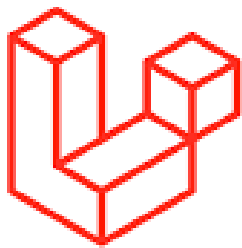


# Projecte GLOBAL: VideosApp

1r SPRINT



# Laravel

**Professor:** Jordi Vega


**Assignatures:** M7, M8 i M9

**Nom:** Harvey

**Cognoms:** John Glover

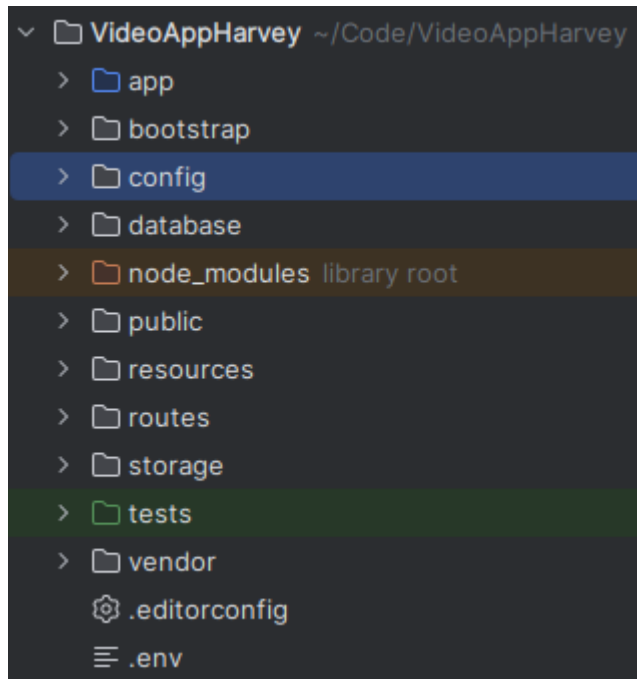
# INDEX:

<b>.env</b>	<b>3</b>
<b>Helpers.php</b>	<b>4</b>
<b>UserHelper.php</b>	<b>5</b>
<b>HelpersTest.php</b>	<b>6</b>
<b>DatabaseSeeder.php</b>	<b>8</b>
<b>Base de Dades (Users)</b>	<b>10</b>
Taula de users:	10
Taula de teams:	10
<b>Procés d'errors</b>	<b>11</b>
Error 1:	11
Solució 1:	11
Error 2:	12
Solució 2:	12

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

## .env

A la següent ruta veurem el fitxer de **.env**




Estableixo la **informació** per crear els usuaris per defecte i també assignarli la info del equip.

```

DEFAULT_USER_NAME=alumne
DEFAULT_USER_EMAIL=alumne@gmail.com
DEFAULT_USER_PASSWORD=Admin123
DEFAULT_USER_TEAM_ID=1
DEFAULT_USER_TEAM_NAME="Team 1"
DEFAULT_USER_USER_ID=1
DEFAULT_USER_PERSONAL_TEAM=true

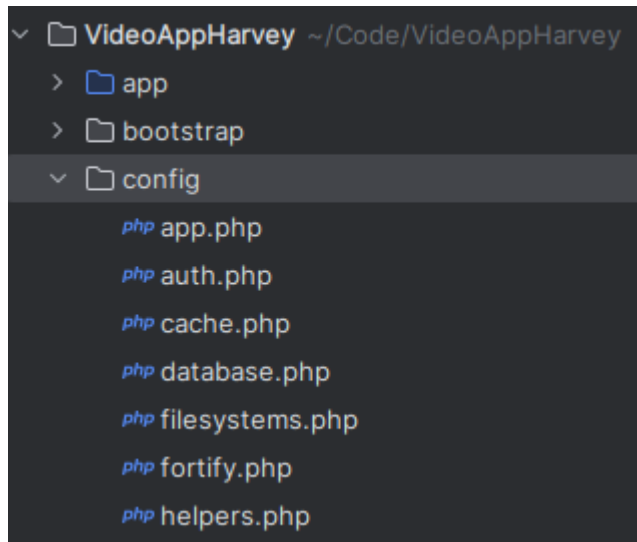
DEFAULT_PROFESSOR_NAME=harvey
DEFAULT_PROFESSOR_EMAIL=harvey@gmail.com
DEFAULT_PROFESSOR_PASSWORD=Admin123
DEFAULT_PROFESSOR_TEAM_ID=2
DEFAULT_PROFESSOR_TEAM_NAME="Team 2"
DEFAULT_PROFESSOR_USER_ID=2
DEFAULT_PROFESSOR_PERSONAL_TEAM=true

```

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

## Helpers.php

A la següent ruta veurem el fitxer de **Helpers.php**

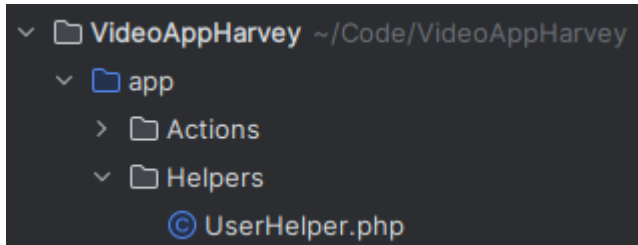


Al següent fitxer he de tindre la mateixa info que el env que se guarde en **variables**.

```
return [
    'default_user' => [
        'name' => env( key: 'DEFAULT_USER_NAME', default: 'alumne'),
        'email' => env( key: 'DEFAULT_USER_EMAIL', default: 'alumne@gmail.com'),
        'password' => env( key: 'DEFAULT_USER_PASSWORD', default: 'Admin123'),
        'team_id' => env( key: 'DEFAULT_USER_TEAM_ID', default: 1),
        'team_name' => env( key: 'DEFAULT_USER_TEAM_NAME', default: 'Team 1'),
        'user_id' => env( key: 'DEFAULT_USER_USER_ID', default: 1),
        'personal_team' => env( key: 'DEFAULT_USER_PERSONAL_TEAM', default: true),
    ],
    'default_professor' => [
        'name' => env( key: 'DEFAULT_PROFESSOR_NAME', default: 'harvey'),
        'email' => env( key: 'DEFAULT_PROFESSOR_EMAIL', default: 'harvey@gmail.com'),
        'password' => env( key: 'DEFAULT_PROFESSOR_PASSWORD', default: 'Admin123'),
        'team_id' => env( key: 'DEFAULT_PROFESSOR_TEAM_ID', default: 2),
        'team_name' => env( key: 'DEFAULT_PROFESSOR_TEAM_NAME', default: 'Team 2'),
        'user_id' => env( key: 'DEFAULT_PROFESSOR_USER_ID', default: 2),
        'personal_team' => env( key: 'DEFAULT_PROFESSOR_PERSONAL_TEAM', default: true),
    ],
];
```

# UserHelper.php

A la següent ruta veurem el fitxer de **UserHelper.php**



Creo una funció anomenada `createDefaultUser` que fa el següent:

- Guarda les dades de default user del fitxer de helper a la variable **userData**
- Crea o troba si ja està creat l'id i info del **equip**
- Crea l'usuari per defecte amb la info que té la **variable** guardada prèviament
- Actualitza el propietari de l'**equip**
- **retorna** l'usuari a la funció


```
public static function createDefaultUser()
{
    $userData = config('key: 'helpers.default_user');

    // Crear o trobar l'equip amb l'ID especificat
    $team = Team::firstOrCreate(
        ['id' => $userData['team_id']], // Usar l'ID de l'equip de la configuració
        ['name' => 'Team ' . $userData['team_id'], 'user_id' => $userData['user_id'], 'personal_team' => true]
    );

    // Crear l'usuari per defecte i assignar-li l'equip
    $user = User::create([
        'name' => $userData['name'],
        'email' => $userData['email'],
        'password' => Hash::make($userData['password']),
        'current_team_id' => 1,
    ]);

    // Actualitzar el propietari de l'equip
    $team->update(['user_id' => $user->id]);

    return $user;
}
```

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

Creo una funció anomenada `createDefaultProfessor` que fa el següent:

- Guarda les dades de default professor del fitxer de helper a la variable **professorData**
- Crea o troba si ja està creat l'id i info del **equip**
- Crea l'usuari per defecte amb la info que té la **variable** guardada prèviament
- Actualitza el **propietari** de l'equip
- **retorna** l'usuari a la funció

```
public static function createDefaultProfessor()
{
    $professorData = config('key: 'helpers.default_professor');

    // Crear o trobar l'equip amb l'ID especificat
    $team = Team::firstOrCreate(
        ['id' => $professorData['team_id']], // Usar l'ID de l'equip de la configuració
        ['name' => 'Team ' . $professorData['team_id'], 'user_id' => $professorData['user_id'], 'personal_team' => true]
    );

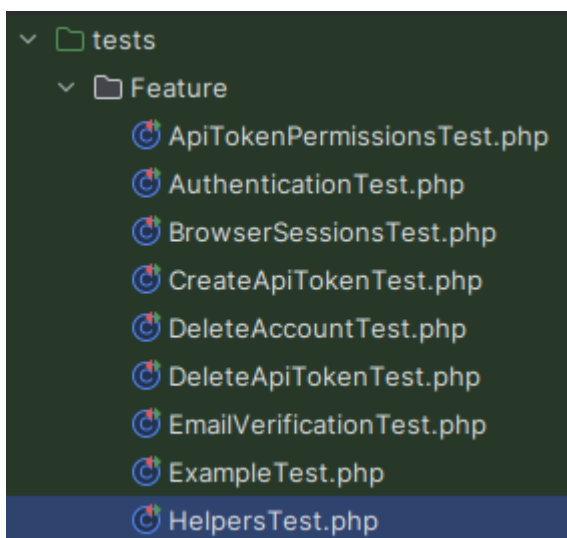
    // Crear el professor per defecte i assignar-li l'equip
    $professor = User::create([
        'name' => $professorData['name'],
        'email' => $professorData['email'],
        'password' => Hash::make($professorData['password']),
        'current_team_id' => 2,
    ]);


    // Actualitzar el propietari de l'equip
    $team->update(['user_id' => $professor->id]);

    return $professor;
}
```

## HelpersTest.php

A la següent ruta veurem el fitxer de **HelpersTest.php**



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

Creo una funció anomenada `crea_els_usuaris_per_defecte` que creara i comprobarà que s'hagi creat els dos usuaris per defecte de manera separada als que és guardan al sistema.

La funció del test fa el següent:


- Creo els dos usuaris cridant les dos funcions del **UserHelper**
- Estableixo la info dels usuaris i les assigno a **variables** per comprovar el resultat obtingut de les funcions prèvies
- El **RefreshDatabase** es borrarà la info una vegada el test hagi finalitzat

```
class HelpersTest extends TestCase
{
    use RefreshDatabase;

    #[Test] // Use the PHPUnit attribute instead of @test
    public function crea_els_usuaris_per_defecte()
    {
        // Creació d'usuaris
        $user = UserHelper::createDefaultUser();
        $professor = UserHelper::createDefaultProfessor();

        // Default User
        $this->assertInstanceOf( expected: User::class, $user);
        $this->assertEquals( expected: 1, $user->current_team_id);
        // Verifiquem que l'usuari té l'equip 1
        $this->assertDatabaseHas( table: 'teams', [
            'id' => 1,
            'user_id' => 1,
            'name' => 'Team 1',
            'personal_team' => 1,
        ]);

        // Default Professor
        $this->assertInstanceOf( expected: User::class, $professor);
        $this->assertEquals( expected: 2, $professor->current_team_id);
        // Verifiquem que el professor té l'equip 2
        $this->assertDatabaseHas( table: 'teams', [
            'id' => 2,
            'user_id' => 2,
            'name' => 'Team 2',
            'personal_team' => 1,
        ]);
    }
}
```

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

Executare el **test** i veure que funciona i es crea els usuaris, i gràcies

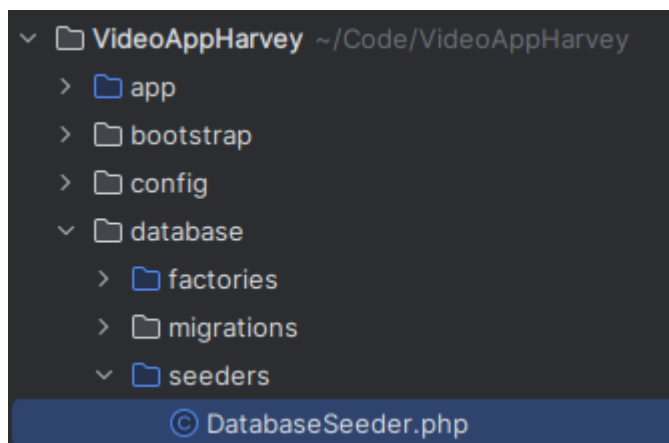
```
alumnat@harvey:~/Code/VideoAppHarvey$ php artisan test --filter=HelpersTest

PASS Tests\Feature\HelpersTest
✓ crea els usuaris per defecte


Tests: 1 passed (6 assertions)
Duration: 0.29s
```

## DatabaseSeeder.php

A la següent ruta veurem el fitxer de **DatabaseSeeder.php**





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

Per a que és quedin creats el usuaris per defecte al sistema, haurem de cridar les funcions al fitxer de **DatabaseSeeder** i una vegada el executem, és quedaran creats.

```
class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     */
    public function run(): void
    {
        // User::factory(10)->withPersonalTeam()->create();
        //User::factory()->withPersonalTeam()->create([
        //    'name' => 'Test User',
        //    'email' => 'test@example.com',
        //]);

        // Create the default user and professor
        UserHelper::createDefaultUser();
        UserHelper::createDefaultProfessor();
    }
}
```

Finalment executarem la comanda **[php artisan db:seed]** per deixarlos creats ha que s'executa el fitxer que executa les funcions

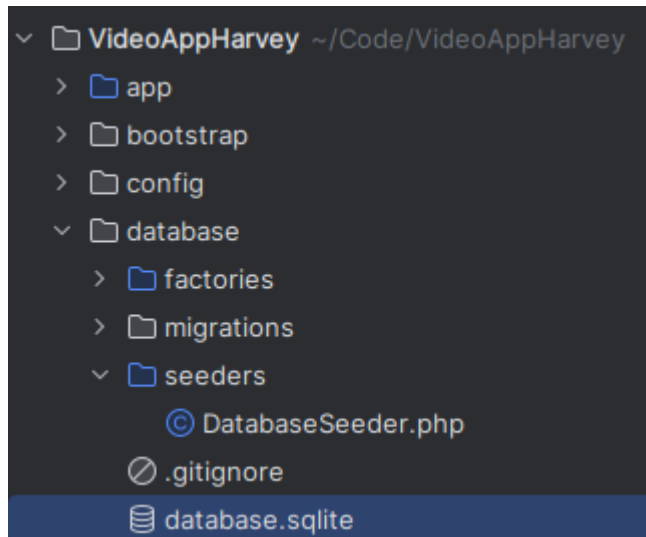
```
alumnat@harvey:~/Code/VideoAppHarvey$ php artisan db:seed

INFO Seeding database.

alumnat@harvey:~/Code/VideoAppHarvey$
```

## Base de Dades (Users)

A la següent ruta veurem el fitxer de **database.sqlite**



### Taula de users:

Veurem que s'ha plenat la info correctament amb id, nom, correu, team i una contrasenya encriptada amb el hash que bé per defecte amb laravel.

users					
id	name	email	current_team_id	password	
1	alumne	alumne@gmail.com	1	\$2y\$12\$.anL4/ovAZnq4wLq0trA1eytJ.JoqqiCaMftLzNU6n1aVkh4LRKV2	
2	harvey	harvey@gmail.com	2	\$2y\$12\$HYLyC2nw9Nq5ULJzbXrV3eJ1tI06PxPY73Bpz/jtcgYYQed/HDnw2	

### Taula de teams:

Veurem que s'ha plenat la info correctament amb id, user\_id, nom, personal team en aquest cas és 1 ja que és true.

teams						
id	user_id	name	personal_team	created_at	updated_at	
1	1	Team 1	1	2024-11-22 18:58:59	2024-11-22 18:58:59	
2	2	Team 2	1	2024-11-22 18:58:59	2024-11-22 18:58:59	

## Procés d'errors

### Error 1:

El primer error era culpa de el atribut deprecated per una actualització recent.

```
class HelpersTest extends TestCase
{
    use RefreshDatabase;

    /** @test */
    public function crea_els_usuaris_per_defecte()
    {
```

Aquí veurem el warning.

```
alumnat@harvey:~/Code/VideoAppHarvey$ php artisan test --filter=HelpersTest

WARN Metadata found in doc-comment for method Tests\Feature\HelpersTest::crea_els_usuaris_per_defecte()
ibutes instead.
```

### Solució 1:

Per solucionar aquest error, haurem de posar el #[Test] ja que el atribut és així a l'actualització més recent.

```
class HelpersTest extends TestCase
{
    use RefreshDatabase;


    #[Test] // Use the PHPUnit attribute instead of @test
    public function crea_els_usuaris_per_defecte()
    {
```

```
alumnat@harvey:~/Code/VideoAppHarvey$ php artisan test --filter=HelpersTest

PASS Tests\Feature\HelpersTest
✓ crea els usuaris per defecte

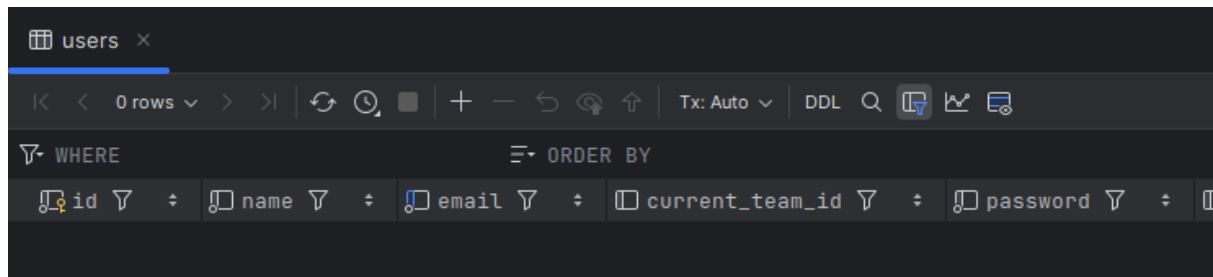
Tests: 1 passed (6 assertions)
Duration: 0.29s

alumnat@harvey:~/Code/VideoAppHarvey$
```

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

## Error 2:

Quan executaba el test em esborrava el contingut una vegada finalitzat el test.



## Solució 2:

He mogut la crida de les funcions de crear a el fitxer de DatabaseSeeder.

```
public function run(): void
{
    // User::factory(10)->withPersonalTeam()->create();
    //User::factory()->withPersonalTeam()->create([
    //    'name' => 'Test User',
    //    'email' => 'test@example.com',
    //]);

    // Create the default user and professor
    UserHelper::createDefaultUser();
    UserHelper::createDefaultProfessor();
}
```

Executo el seeder i ja están creats i es queden creats.

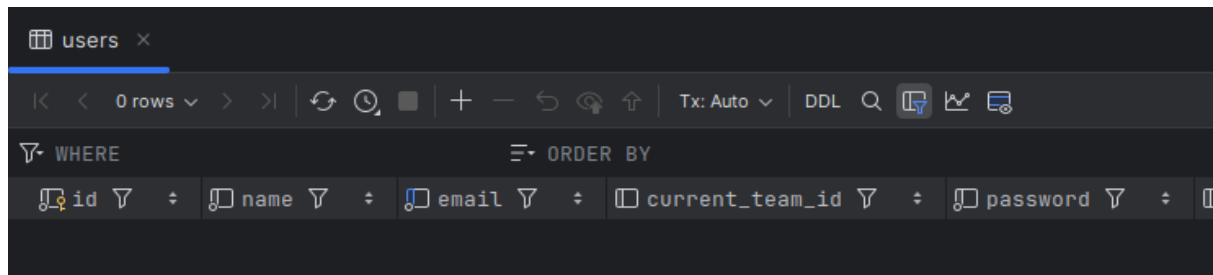
```
alumnat@harvey:~/Code/VideoAppHarvey$ php artisan db:seed

INFO Seeding database.

alumnat@harvey:~/Code/VideoAppHarvey$
```

### Error 3:

Després d'executar el test, seguia esborrant el contingut de les taules.



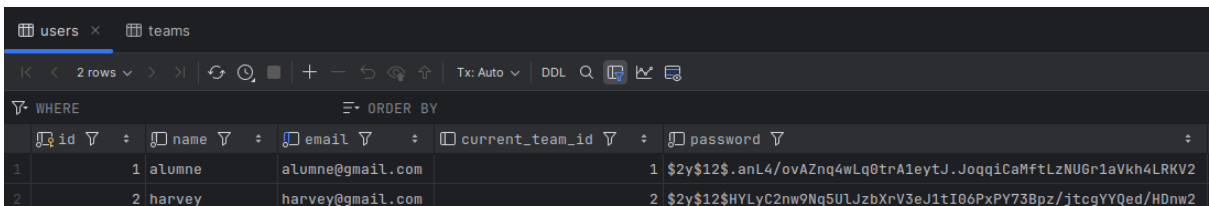
Veurem al fitxer de phpunit.xml hi han 2 línies comentades.

```
<php>
  <env name="APP_ENV" value="testing"/>
  <env name="APP_MAINTENANCE_DRIVER" value="file"/>
  <env name="BCRYPT_ROUNDS" value="4"/>
  <env name="CACHE_STORE" value="array"/>
  <!-- <env name="DB_CONNECTION" value="sqlite"/> -->
  <!-- <env name="DB_DATABASE" value=":memory:"/> -->
  <env name="MAIL_MAILER" value="array"/>
```


### Solució 3:

Descomento les línies i ja em queda guardat el contingut després d'executar el test

```
<php>
  <env name="APP_ENV" value="testing"/>
  <env name="APP_MAINTENANCE_DRIVER" value="file"/>
  <env name="BCRYPT_ROUNDS" value="4"/>
  <env name="CACHE_STORE" value="array"/>
  <env name="DB_CONNECTION" value="sqlite"/>
  <env name="DB_DATABASE" value=":memory:"/>
```



	id	name	email	current_team_id	password
1	1	alumne	alumne@gmail.com	1	\$2y\$12\$.anL4/ovAZnq4wLq0trA1eytJ.JoqqiCaMftLzNU6r1aVkh4LRKV2
2	2	harvey	harvey@gmail.com	2	\$2y\$12\$HYLyC2nw9Nq5ULJzbXrV3eJ1tI06PxPY73Bpz/jtcgYYQed/HDnw2

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

## Error 4:

Un error comú que tenia era confusió de variables ja que algunes eren variables del helper i altres de la funció on es crea l'usuari.

**helpers:** team\_id

```
'default_user' => [
  'name' => env( key: 'DEFAULT_USER_NAME', default: 'alumne'),
  'email' => env( key: 'DEFAULT_USER_EMAIL', default: 'alumne@gmail.com'),
  'password' => env( key: 'DEFAULT_USER_PASSWORD', default: 'Admin123'),
  'team_id' => env( key: 'DEFAULT_USER_TEAM_ID', default: 1),
```

**userhelper:** id

```
$this->assertDatabaseHas( table: 'teams', [
  'id' => 1,
  'user_id' => 1,
  'name' => 'Team 1',
  'personal_team' => 1,
]);
```

Error de variables:

```
$team = Team::firstOrCreate(
  ['id' => $userData['id']] // Usar l'ID de l'equip de la configuració
  ['name' => 'Team ' . $userData['team_id'], 'user_id' => $userData['user_id'], 'personal_team' => true]
);
```

## Solució 4:

La solució era assegurar-me en quin ordre usar les variables per a que no doni error.

```
$team = Team::firstOrCreate(
  ['id' => $userData['team_id']] // Usar l'ID de l'equip de la configuració
  ['name' => 'Team ' . $userData['team_id'], 'user_id' => $userData['user_id'], 'personal_team' => true]
);
```