

# Projecte LaravelTinkering



**Professor:** Jordi Vega  
**Assignatura:** M9 Programació de serveis i processos  
**Nom:** Harvey  
**Cognoms:** John Glover



# INDEX:

|                  |    |
|------------------|----|
| HEADER/NAVBAR    | 4  |
| FOOTER           | 6  |
| BASE DE DADES    | 7  |
| CONFIGS GENERALS | 11 |
| LANDING PAGE     | 24 |
| FILMS            | 26 |
| CARS             | 35 |
| RESPONSIVITAT    | 45 |



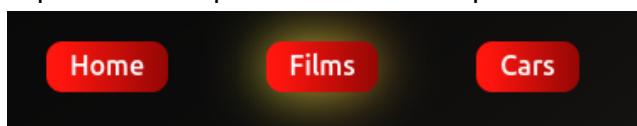
# HEADER/NAVBAR

## RESULTAT:

El header està compost de 3 apartats que son Home, Films i Cars.



Si paso el ratolí per damunt de les opcions és iluminen.



## CODI:

Cada opció del menú està configurat per anar a una pàgina en específic i quan està en mode mobil està configurat per tindre un popup amb les opcions, i les altres desapareixen.

```
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Projecte Laravel</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">
    <link href="https://fonts.googleapis.com/css2?family=Chakra+Petch:wght@700&display=swap" rel="stylesheet">
    <script src="https://cdnjs.cloudflare.com/ajax/libs/three.js/r128/three.min.js"></script>
    <script src="https://cdn.jsdelivr.net/npm/dat.gui@0.7.7/build/dat.gui.min.js"></script>
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css" rel="stylesheet">
    <link href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css" rel="stylesheet">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">
</head>
```

```
<div class="flex lg:flex-1">
    <a href="/" class="-m-1.5 p-1.5">
        <span class="sr-only">Harvey</span>
        
    </a>
</div>
```

```
<div class="hidden lg:flex lg:gap-x-12">
    <div class="button-container">
        <a href="{{ route('home') }}" class="text-lg font-semibold text-white px-6 py-3">Home</a>
    </div>
    <div class="button-container">
        <a href="{{ route('films.index') }}" class="text-lg font-semibold text-white px-6 py-3">Films</a>
    </div>
    <div class="button-container">
        <a href="{{ route('cars.index') }}" class="text-lg font-semibold text-white px-6 py-3">Cars</a>
    </div>
</div>
```



Està és la funció que desplega el menú en mode mòbil, canvia el icono i amaga el contingut quan no és necessari.

```
<script>
    // Get the mobile menu and buttons
    const menuButton = document.getElementById('menuButton');
    const mobileMenu = document.getElementById('mobileMenu');
    const closeMenuButton = document.getElementById('closeMenu');

    // Show the mobile menu when the menu button is clicked
    menuButton.addEventListener('click', function() {
        mobileMenu.classList.remove('hidden');
    });

    // Close the mobile menu when the close button is clicked
    closeMenuButton.addEventListener('click', function() {
        mobileMenu.classList.add('hidden');
    });

    // Close the menu when clicking anywhere outside the menu
    window.addEventListener('click', function(event) {
        if (!mobileMenu.contains(event.target) && !menuButton.contains(event.target)) {
            mobileMenu.classList.add('hidden');
        }
    });

    // Function to close menu after selecting an option
    function closeMobileMenu() {
        mobileMenu.classList.add('hidden');
    }
</script>
```



# FOOTER

## RESULTAT:

Al footer he fet un disseny simple de Tots drets reservats i uns fons transparent per ajustarse a qualsevol pagina.

© 2024 Harvey. All Rights Reserved.

## CODI:

Al codi s'observa que està centrat i amb el tipus de lletra, color, mida, etc...

```
<footer class="bg-black text-white py-4 text-center" style="...">
    <p>© 2024 Harvey. All Rights Reserved.</p>
</footer>
```

# BASE DE DADES

Database migrations:

Crearem el fitxer.

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

INFO Migration [database/migrations/2024_12_19_084218_create_films_table.php] created successfully.
```

Dins crearem la taula amb els seus camps.

```
php 2024_12_19_084218_create_films_table.php ×
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration {
8     ▲ Wharvey123
9     public function up()
10    {
11         Schema::create('films', function (Blueprint $table) {
12             $table->id(); // ID autoincremental
13             $table->string('name'); // Nom de la pel·lícula
14             $table->string('director'); // Nom del director
15             $table->integer('year'); // Any de producció
16             $table->text('description'); // Descripció
17             $table->timestamps(); // Camps created_at i updated_at
18        });
19
20        ▲ Wharvey123
21        public function down()
22        {
23            Schema::dropIfExists('films');
24        };
25
26    }
```

Crearem el fitxer.

```
alumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan make:model Car -m

INFO Model [app/Models/Car.php] created successfully.

INFO Migration [database/migrations/2024_12_19_084208_create_cars_table.php] created successfully.
```

Dins crearem la taula amb els seus camps.

```
php 2024_12_19_084208_create_cars_table.php ×

1  <?php
2
3  use Illuminate\Database\Migrations\Migration;
4  use Illuminate\Database\Schema\Blueprint;
5  use Illuminate\Support\Facades\Schema;
6
7  return new class extends Migration {
8      ↑ Wharvey123
9      public function up()
10     {
11         Schema::create('cars', function (Blueprint $table) {
12             $table->id(); // ID autoincremental
13             $table->string('make'); // Marca
14             $table->string('model'); // Model
15             $table->integer('year'); // Any de producció
16             $table->float('price', precision: 10, 2); // Preu amb decimals
17             $table->text('description'); // Descripció
18             $table->timestamps(); // Camps created_at i updated_at
19         });
20
21         ↑ Wharvey123
22         public function down()
23         {
24             Schema::dropIfExists('cars');
25         }
26     }
```

Executare el migrate.

```
alumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan migrate

INFO Running migrations.

2024_12_19_084208_create_cars_table ...
2024_12_19_084218_create_films_table ...
```

## TAULA (films):

A la taula de films he creat 7 camps (id, nom, director, any, descripció, created\_at i updated\_at) amb el tipus adequat per a cada camp.

He procurat que el id sigui de tipus autoincrement així sé que muntarà de valor cada vegada que és cree un nou.

| ▼ | .films                      |
|---|-----------------------------|
| ▼ | columns 7                   |
| ↳ | id integer (auto increment) |
| ↳ | name varchar                |
| ↳ | director varchar            |
| ↳ | year integer                |
| ↳ | description text            |
| ↳ | created_at datetime         |
| ↳ | updated_at datetime         |

Una vegada planada la taula quedaria així:

| WHERE |                   | ORDER BY             |      |
|-------|-------------------|----------------------|------|
|       |                   |                      |      |
| 1     | 1 Inception       | Christopher Nolan    | 2010 |
| 2     | 2 The Godfather   | Francis Ford Coppola | 1972 |
| 3     | 3 Parasite        | Bong Joon Ho         | 2019 |
| 4     | 4 The Dark Knight | Christopher Nolan    | 2008 |
| 5     | 5 Pulp Fiction    | Quentin Tarantino    | 1994 |

| description                                | created_at          | updated_at          |
|--|---------------------|---------------------|
| A mind-bending thriller about dream inv... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| A classic tale of family, power, and be... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| A gripping social satire that took the ... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| A groundbreaking superhero film about B... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| An iconic crime drama with intertwined ... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |

## TAULA (cars):

A la taula de cars he creat 8 camps (id, marca, model, any, preu, descripció, created\_at i updated\_at) amb el tipus adequat per a cada camp.

He procurat que el id sigui de tipus autoincrement així sé que muntarà de valor cada vegada que és cree un nou.

| cars        |                          |
|-------------|--------------------------|
|             | columns 8                |
| id          | integer (auto increment) |
| make        | varchar                  |
| model       | varchar                  |
| year        | integer                  |
| price       | numeric                  |
| description | text                     |
| created_at  | datetime                 |
| updated_at  | datetime                 |

Una vegada planada la taula quedaria així:

|   | id | make      | model     | year | price    |
|---|----|-----------|-----------|------|----------|
| 1 | 1  | Toyota    | Corolla   | 2021 | 20000    |
| 2 | 2  | Ford      | Mustang   | 2020 | 35000    |
| 3 | 3  | Tesla     | Model S   | 2022 | 79999.99 |
| 4 | 4  | Honda     | Civic     | 2019 | 18000.5  |
| 5 | 5  | Chevrolet | Silverado | 2023 | 45000    |

| description                                | created_at          | updated_at          |
|--|---------------------|---------------------|
| Reliable and fuel-efficient sedan, perf... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| Iconic sports car with powerful perform... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| Luxury electric sedan with cutting-edge... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| Compact car known for its durability an... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |
| Powerful pickup truck built for tough j... | 2024-12-22 14:52:37 | 2024-12-22 14:52:37 |

# CONFIGS GENERALS

## app.blade.php

Procuro cridar el css i js al header. Després incloure el header i footer i això s'aplica a totes les pàgines.

```
app.blade.php ×

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>LaravelTinkering</title>
7      @vite(['resources/css/app.css'])
8      @vite(['resources/js/app.js'])
9  </head>
10 <body class="bg-black">
11
12 <!-- Header -->
13 @include('layouts.header')
14
15 <!-- Contingut de la pàgina -->
16 <main class="flex-1">
17     @yield('content')
18 </main>
19
20 <!-- Footer -->
21 @include('layouts.footer')
22
23 </body>
24 </html>
25 |
```

## web.php:

Al document de web.php he afegit les rutes de cada pàgina films i cars, en aquest cas son create, store, show, delete, update, destroy i edit.

Ja que les funcions com delete son específics al id els he diferenciat amb la ruta posant el nom de la apartat davant com ara films o cars.

```
php web.php ×  
1  <?php  
2  
3  use Illuminate\Support\Facades\Route;  
4  use App\Http\Controllers\FilmController;  
5  use App\Http\Controllers\CarController;  
6  
7  // Ruta principal - Ruta per la pàgina d'inici  
8  Route::get('uri: /', function () {  
9      return view('view: index');  
10 })->name('name: home'); // Ruta anomenada 'home'  
11  
12 // Rutes de recursos per a Films  
13 // La següent línia genera totes les rutes per al FilmController automàticament:  
14 // - index, create, store, show, edit, update, destroy  
15 Route::resource('name: films', 'controller: FilmController::class'); // Utilitzant Route::resource() per al FilmController  
16 // Ruta personalitzada per a la confirmació d'eliminació  
17 Route::get('uri: /films/{film}/delete', [FilmController::class, 'confirmDelete'])->name('name: films.confirmDelete');  
18  
19 // Rutes de recursos per a Cars  
20 // La següent línia genera totes les rutes per al CarController automàticament:  
21 // - index, create, store, show, edit, update, destroy  
22 Route::resource('name: cars', 'controller: CarController::class'); // Utilitzant Route::resource() per al CarController  
23 // Ruta personalitzada per a la confirmació d'eliminació  
24 Route::get('uri: /cars/{car}/delete', [CarController::class, 'confirmDelete'])->name('name: cars.confirmDelete');  
25 |
```



## FilmSeeder.php:

Crear el seeder.

```
alumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan make:seeder FilmSeeder
INFO  Seeder [database/seeders/FilmSeeder.php] created successfully.
```

Dins del seeder creo 5 inserts per a la taula correspondent en aquest cas films.

```
© FilmSeeder.php ×
1 <?php
2
3 namespace Database\Seeders;
4
5 use Illuminate\Database\Seeder;
6 use Illuminate\Support\Facades\DB;
7
8 usage  ↳ Wharvey123 *
9 class FilmSeeder extends Seeder
{
    ↳ Wharvey123 *
10    public function run()
11    {
12        DB::table('films')->insert([
13            [
14                'name' => 'Inception',
15                'director' => 'Christopher Nolan',
16                'year' => 2010,
17                'description' => 'A mind-bending thriller about dream invasion.',
18                'created_at' => now(),
19                'updated_at' => now(),
20            ],
21        ]);
22    }
}
```

## CarSeeder.php:

Crear el seeder.

```
alumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan make:seeder CarSeeder
INFO  Seeder [database/seeders/CarSeeder.php] created successfully.
```

Dins del seeder creo 5 inserts per a la taula correspondent en aquest cas cars.

```
② CarSeeder.php ×
1  <?php
2
3  namespace Database\Seeders;
4
5  use Illuminate\Database\Seeder;
6  use Illuminate\Support\Facades\DB;
7
8  usage  ▲ Wharvey123 *
9  class CarSeeder extends Seeder
{
    ▲ Wharvey123 *
10     public function run()
11     {
12         DB::table('cars')->insert([
13             [
14                 'make' => 'Toyota',
15                 'model' => 'Corolla',
16                 'year' => 2021,
17                 'price' => 20000.00,
18                 'description' => 'Reliable and fuel-efficient sedan, perfect for daily commuting.',
19                 'created_at' => now(),
20                 'updated_at' => now(),
21             ],
22         ]);
23     }
}
```

Executo el db:seed per aplicar els canvis i inserir els camps.

```
alumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan db:seed
INFO  Seeding database.

Database\Seeders\FilmSeeder .....
Database\Seeders\FilmSeeder .....

Database\Seeders\CarSeeder .....
Database\Seeders\CarSeeder .....
```



## FilmController.php:

Creo el controlador.

```
aLumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan make:controller FilmController --resource
INFO Controller [app/Http/Controllers/FilmController.php] created successfully.
```

Al filmcontroller començó posant la ruta del fitxer i obtenint totes les pelis.

```
© FilmController.php ×
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Film;
6 use Illuminate\Http\Request;
7
8 * Wharvey123 *
9 class FilmController extends Controller
10 {
11     * Wharvey123
12     public function index()
13     {
14         $films = Film::all();
15         return view( view: 'films.index', compact( var_name: 'films'));
16     }
17 }
```

Ara he modificat el create, store, show, update, destroy i confirmdelete per anar a la ruta correspondient de films con ara films.algo.

Create

```
* Wharvey123
public function create()
{
    return view( view: 'films.create');
}
```

## Store

```
public function store(Request $request)
{
    $request->validate([
        'name' => 'required|string|max:255',
        'director' => 'required|string|max:255',
        'year' => 'required|integer|min:1800|max:' . date('Y'),
        'description' => 'nullable|string',
    ]);

    Film::create($request->all());
    return redirect()->route('films.index')->with('success', 'Film created successfully.');
}
```

## Show

```
public function show(string $id)
{
    $film = Film::findOrFail($id);
    return view('films.show', compact('film'));
}

▲ Wharvey123
public function edit(string $id)
{
    $film = Film::findOrFail($id);
    return view('films.edit', compact('film'));
}
```

## ConfirmDelete

```
public function confirmDelete(string $id)
{
    $film = Film::findOrFail($id);
    return view('films.delete', compact('film'));
}
```



## Update

```
public function update(Request $request, string $id)
{
    $request->validate([
        'name' => 'required|string|max:255',
        'director' => 'required|string|max:255',
        'year' => 'required|integer|min:1800|max:' . date('Y'),
        'description' => 'nullable|string',
    ]);

    $film = Film::findOrFail($id);
    $film->update($request->all());
    return redirect()->route('films.index')->with('success', 'Film updated successfully.');
}
```

## Destroy

```
public function destroy(Request $request, string $id)
{
    $film = Film::findOrFail($id);

    // Protegir contra CSRF i assegurar-se que és una petició DELETE
    if ($request->isMethod('delete')) {
        $film->delete();
        return redirect()->route('films.index')->with('success', 'Film deleted successfully.');
    }

    return redirect()->route('films.index')->with('error', 'Invalid delete request.');
}
```



## CarController.php:

Creo el controlador.

```
alumnat@harvey:~/Code/M7-8-9/laravelTinkering$ php artisan make:controller CarController --resource
INFO Controller [app/Http/Controllers/CarController.php] created successfully.
```

Al carcontroller he fet algo molt similar al films cridant tots els cotxes i després indicant el create, store, show, update, destroy i confirmdelete per anar a la ruta correspondient de films con ara cars.algo.

```
© CarController.php ×
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Car;
6 use Illuminate\Http\Request;
7
8 /**
9 * @var \App\Models\Car
10 */
11 class CarController extends Controller
12 {
13     /**
14      * @var \App\Models\Car
15     */
16     public function index()
17     {
18         $cars = Car::all(); // Fetch all cars
19         return view('cars.index', compact('cars'));
20     }
21 }
```

Create

```
public function create()
{
    return view('cars.create'); // Render the create car form
}
```

## Store

```
public function store(Request $request)
{
    $request->validate([
        'make' => 'required|string|max:255',
        'model' => 'required|string|max:255',
        'year' => 'required|integer|min:1800|max:' . date('Y'),
        'price' => 'required|numeric|min:0',
        'description' => 'nullable|string',
    ]);

    Car::create($request->all()); // Save the car data
    return redirect()->route('cars.index')->with(['success' => 'Car created successfully.']);
}
```

## Show

```
public function show(string $id)
{
    $car = Car::findOrFail($id); // Find the car or throw a 404 error
    return view('cars.show', compact('car')); // Pass the single car to the view
}
```

## Edit

```
public function edit(string $id)
{
    $car = Car::findOrFail($id); // Find the car for editing
    return view('cars.edit', compact('car')); // Pass the car data to the view
}
```

## Update

```
public function update(Request $request, string $id)
{
    $request->validate([
        'make' => 'required|string|max:255',
        'model' => 'required|string|max:255',
        'year' => 'required|integer|min:1800|max:' . date('Y'),
        'price' => 'required|numeric|min:0',
        'description' => 'nullable|string',
    ]);

    $car = Car::findOrFail($id); // Find the car to update
    $car->update($request->all()); // Update car data
    return redirect()->route('cars.index')->with('success', 'Car updated successfully.');
}
```

## ConfirmDelete

```
public function confirmDelete(string $id)
{
    $car = Car::findOrFail($id); // Find the car for deletion confirmation
    return view('cars.delete', compact('car')) // Pass the car data to the delete view
}
```

## Destroy

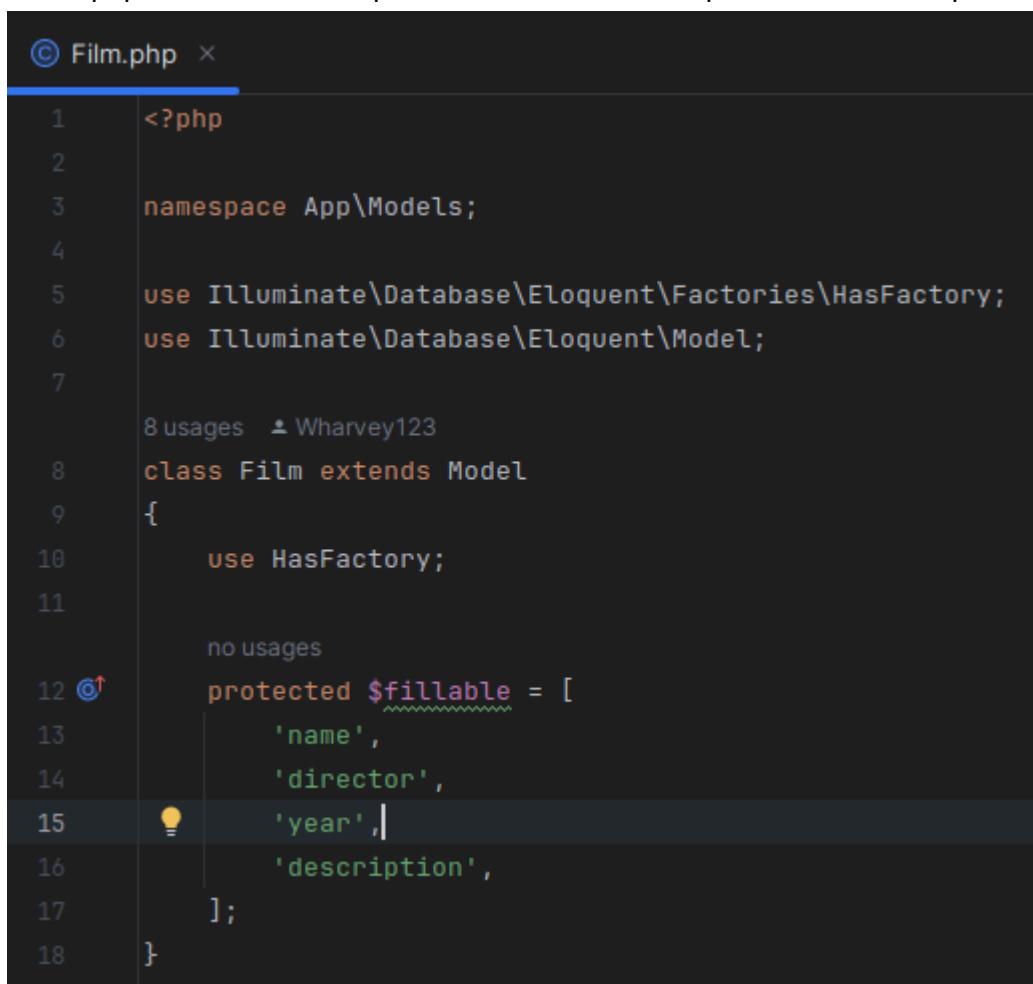
```
public function destroy(Request $request, string $id)
{
    $car = Car::findOrFail($id); // Find the car to delete

    // Protect against CSRF and ensure it is a DELETE request
    if ($request->isMethod('delete')) {
        $car->delete(); // Delete the car
        return redirect()->route('cars.index')->with('success', 'Car deleted successfully.');
    }

    return redirect()->route('cars.index')->with('error', 'Invalid delete request.');
}
```

## Film.php:

Al film.php hi ha una funció que crida tot de la bd i els plena amb els camps correctes.



The screenshot shows a code editor window with the file 'Film.php' open. The code is a PHP class definition for a 'Film' model. It includes namespace declarations, use statements for Illuminate\Database\Eloquent\Factories\HasFactory and Illuminate\Database\Eloquent\Model, and a class definition with methods for HasFactory and fillable. A cursor is visible at the end of the 'year' entry in the fillable array.

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Film extends Model
{
    use HasFactory;

    no usages

    protected $fillable = [
        'name',
        'director',
        'year',
        'description',
    ];
}
```

## Car.php:

Al car.php hi ha una funció que crida tot de la bd i els plena amb els camps correctes.

```
© Car.php ×  
1  <?php  
2  
3  namespace App\Models;  
4  
5  ✓ use Illuminate\Database\Eloquent\Factories\HasFactory;  
6  use Illuminate\Database\Eloquent\Model;  
7  ⚡  
8  8 usages  ↗ Wharvey123  
9  ✓ class Car extends Model  
10 {  
11     use HasFactory;  
12     no usages  
13     protected $fillable = [  
14         'make',  
15         'model',  
16         'year',  
17         'price',  
18         'description',  
19     ];  
20 }
```



## tailwind.config.js:

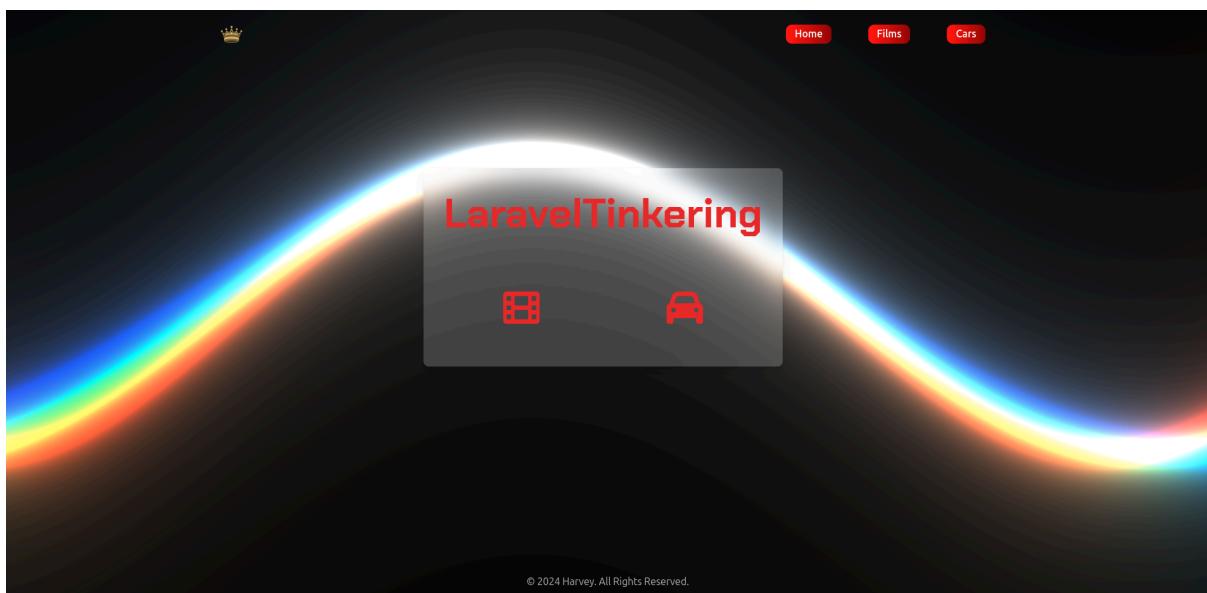
Estableix les rutes, i el lg que és en cas de mòbil, farem els canvis a partir de estos px.

```
js tailwind.config.js ×
1 import defaultTheme from 'tailwindcss/defaultTheme';
2 import forms from '@tailwindcss/forms';
3 import typography from '@tailwindcss/typography';
4
5 /** @type {import('tailwindcss').Config} */
no usages  ↳ Wharvey123 *
6 export default {
7   content: [
8     './resources/views/**/*.blade.php',
9     './resources/js/**/*.js',
10    './vendor/laravel/framework/src/Illuminate/Pagination/resources/views/*.blade.php',
11    './vendor/laravel/jetstream/**/*.blade.php',
12    './storage/framework/views/*.php',
13  ],
14  theme: {
15    extend: {
16      fontFamily: {
17        sans: ['Figtree', ...defaultTheme.fontFamily.sans],
18      },
19    },
20    screens: {
21      'lg': '1024px',
22    },
23  },
24  plugins: [forms, typography],
25};
```

# LANDING PAGE

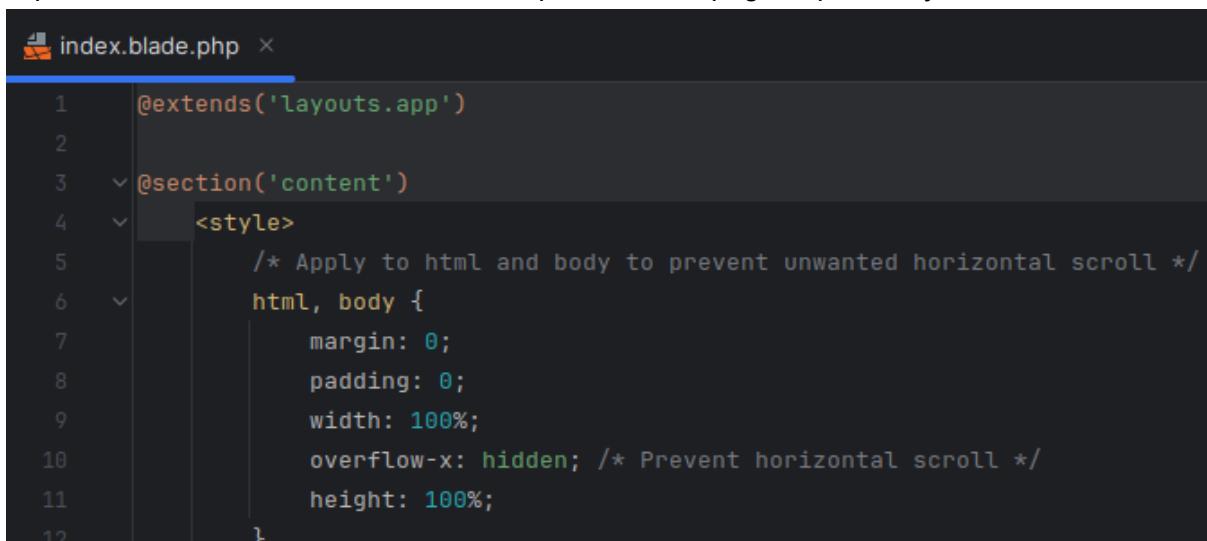
## RESULTAT:

A la landing page tinc un header, footer i al mig de la pantalla un bloc que conté info i 2 botons que es poden clicar per anar a la pàgina que desitges. També efectes de glow de fons i un efecte de hover.



## CODI:

Importo el header i footer i creo 2 botons per anar a la pàgina que desitja el usuari.



```
index.blade.php
```

```
1 @extends('layouts.app')
2
3 @section('content')
4 <style>
5 /* Apply to html and body to prevent unwanted horizontal scroll */
6 html, body {
7     margin: 0;
8     padding: 0;
9     width: 100%;
10    overflow-x: hidden; /* Prevent horizontal scroll */
11    height: 100%;
12}
```



També importo el fons amb el canvas.

```
<!-- WebGL background canvas -->
<canvas id="webgl-canvas"></canvas>

<!-- vertexShader -->
<script id="js-vertex-shader" type="x-shader/x-vertex">
    attribute vec3 position;

    void main() {
        gl_Position = vec4(position, 1.0);
    }
</script>
```

I aquí tindre el container principal.

```
<div class="container mx-auto flex flex-col items-center justify-center min-h-screen text-white content-wrapper">
    <!-- Center the content with flex -->
    <div class="relative bg-white backdrop-blur-md rounded-lg p-8 text-center shadow-lg -translate-y-20">
        <!-- More blurred translucent box and higher positioning -->
        <h1 class="text-6xl font-bold mb-12 leading-tight text-red-600">Laravel Tinkering</h1>
        <!-- Main text size increased and red text applied -->
        <div class="flex justify-center space-x-8">
            <a href="{{ route('films.index') }}">
                <span class="box">
                    <i class="fas fa-film fa-3x text-red-600 sm:fa-2x md:fa-3x"></i>
                </span>
            </a>
            <a href="{{ route('cars.index') }}">
                <span class="box">
                    <i class="fas fa-car fa-3x text-red-600 sm:fa-2x md:fa-3x"></i>
                </span>
            </a>
        </div>
    </div>
</div>
```

# FILMS

## RESULTAT (index):

En aquesta pàgina mostra el contingut de la taula amb opcions de filtrar la info per any, director o individualment per camp en ordre ascendent o descendente.

També i han opcions per afegir, editar, esborrar o observar la pel·lícula en més detall.  
Finalment ho he fet responsive i amb un botó per tornar a la pàgina anterior.

The screenshot shows a dark-themed web application for managing movies. At the top, there's a header with the word "Films" and a back arrow icon. Below the header are two search input fields: one for searching by year ("Cerca per any...") and one for searching by director ("Cerca per director..."). A red button labeled "Afegir Nova Pel·lícula" (Add New Movie) is positioned above the search fields. The main content area displays a table of movie records:

| ID | TÍTOL           | DIRECTOR             | ANY  | ACCIONS |
|----|-----------------|----------------------|------|---------|
| 1  | Inception       | Christopher Nolan    | 2010 |         |
| 2  | The Godfather   | Francis Ford Coppola | 1972 |         |
| 3  | Parasite        | Bong Joon Ho         | 2019 |         |
| 4  | The Dark Knight | Christopher Nolan    | 2008 |         |
| 5  | Pulp Fiction    | Quentin Tarantino    | 1994 |         |

## CODI (index):

Creo el botó de tornar atras i creo la taula amb la funció que tinc en js per ordenar.

```
<div class="flex justify-between items-center mb-8">
  <h1 class="text-5xl font-extrabold title">Films</h1>
  <div class="top-right">
    <a href="/" style="font-size: 36px;" class="text-white" ><i class="fa" style="font-size: 36px;">&#xf137;</i></a>
  </div>
</div>
<div class="top-left mb-4">
  <a href="{{ route('films.create') }}" class="glow-button px-6 py-3 rounded w-full hover:bg-black hover-animate w-full" style="font-size: 16px;">Afegir Nova Pel·lícula</a>
</div>
<div class="filter-container mt-4 mb-4">
  <div class="filter-item">
    <input type="text" id="searchYear" placeholder="Cerca per any..." class="p-2 border border-gray-300 rounded w-full text-black" />
    <i class="fas fa-search"></i>
  </div>
  <div class="filter-item">
    <input type="text" id="searchDirector" placeholder="Cerca per director..." class="p-2 border border-gray-300 rounded w-full text-black" />
    <i class="fas fa-search"></i>
  </div>
</div>
```

```
<div class="table-container mt-12">
  <table class="min-w-full w-full border border-gray-300 rounded-lg overflow-hidden" id="filmsTable">
    <thead>
      <tr class="table-header text-sm uppercase leading-normal">
        <th class="py-3 px-6 text-left" onclick="sortTable(0)">ID <i class="fas fa-sort"></i></th>
        <th class="py-3 px-6 text-left" onclick="sortTable(1)">Titol <i class="fas fa-sort"></i></th>
        <th class="py-3 px-6 text-left" onclick="sortTable(2)">Director <i class="fas fa-sort"></i></th>
        <th class="py-3 px-6 text-left" onclick="sortTable(3)">Any <i class="fas fa-sort"></i></th>
        <th class="py-3 px-6 text-center">Accions</th>
      </tr>
    </thead>
    <tbody class="text-gray-600 text-sm font-light">
      @forelse ($films as $film)
        <tr class="table-row border-b border-gray-200">
          <td class="py-3 px-6" style="border-bottom: none;">{{ $film->id }}{{ $film->name }}
            <a href="{{ route('films.show', $film->id) }}" class="text-blue-500 hover:text-blue-700" style="font-size: 16px;"><i class="fas fa-eye"></i></a>
            <a href="#" class="text-blue-500 hover:text-blue-700" style="font-size: 16px;"><i class="fas fa-pencil-alt"></i></a>
            <a href="#" class="text-blue-500 hover:text-blue-700" style="font-size: 16px;"><i class="fas fa-trash"></i></a>
          </td>
        </tr>
      @empty
        <tr>
          <td colspan="4" style="text-align: center; height: 100px; vertical-align: middle;">No hi ha resultats.
```

Aquí mostro la taula i com mostro cada element

```
<script>
  const searchYearInput = document.getElementById('searchYear'); // Filtrar per any
  const searchDirectorInput = document.getElementById('searchDirector'); // Filtrar per director

  searchYearInput.addEventListener('input', function () {
    const filterYear = searchYearInput.value.toLowerCase();
    filterTableRows(filterYear, 'year');
  });

  searchDirectorInput.addEventListener('input', function () {
    const filterDirector = searchDirectorInput.value.toLowerCase();
    filterTableRows(filterDirector, 'director');
  });
</script>
```

## RESULTAT (create):

Quan afegeixo una nova pel·lícula hi han placeholders que diuen la info que ha de ser una vegada creada és veu creat i afegit a la taula

**Add New Film**

Film Name: Inside Out

Director: Peter Docter

Year: 2015

Description: test

Save Film

Una vegada creada és veu creat i afegit a la taula

**Films**

Afegir Nova Pel·lícula

Cerca per any...

Cerca per director...

| ID | TÍTOL           | DIRECTOR             | ANY  | ACCIONS |
|----|-----------------|----------------------|------|---------|
| 1  | Inception       | Christopher Nolan    | 2010 |         |
| 2  | The Godfather   | Francis Ford Coppola | 1972 |         |
| 3  | Parasite        | Bong Joon Ho         | 2019 |         |
| 4  | The Dark Knight | Christopher Nolan    | 2008 |         |
| 5  | Pulp Fiction    | Quentin Tarantino    | 1994 |         |
| 6  | Inside Out      | Peter Docter         | 2015 |         |

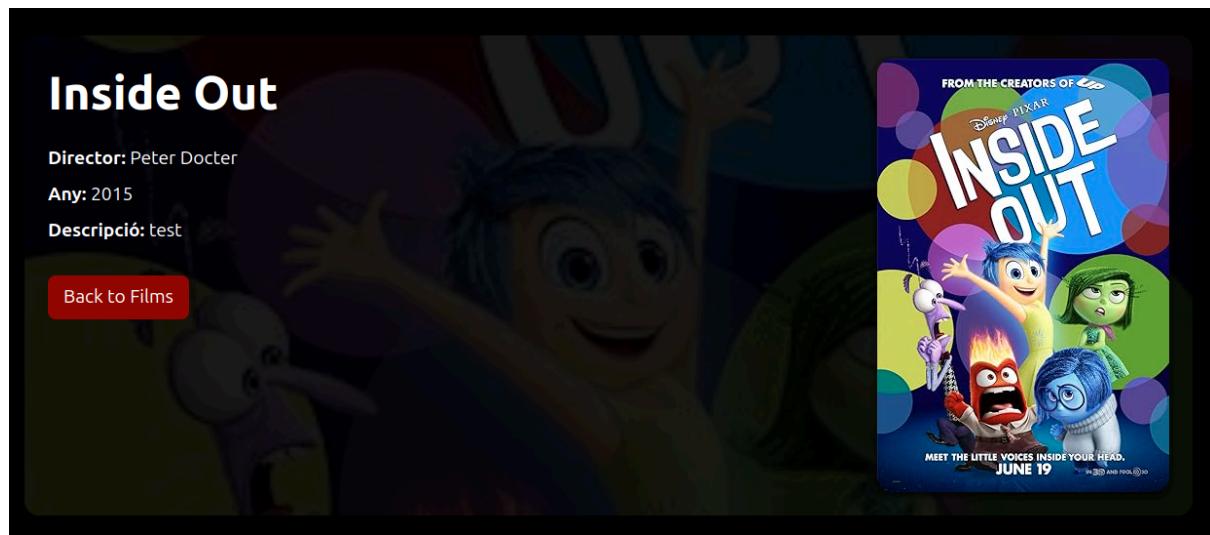
## CODI (create):

Creo el form del create amb els placeholders.

```
<div class="flex items-start justify-center fade-in pt-16 pb-16 px-4 sm:px-16"> <!-- Add responsive padding -->
  <div class="max-w-4xl w-full shadow-lg rounded-lg p-8 bg-white/20 backdrop-blur-md text-white">
    <div class="flex justify-between items-center mb-6">
      <h1 class="text-5xl font-extrabold text-red-600">Add New Film</h1>
      <div>
        <a href="/films" style="..." class="text-red-600 hover:text-white">
          <i class="fa">#xf137;</i>
        </a>
      </div>
    </div>
    <form action="{{ route('films.store') }}" method="POST">
      @csrf
      <!-- Film Name -->
      <div class="mb-4">
        <label for="name" class="block font-semibold">Film Name</label>
        <input type="text" id="name" name="name" required
               class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
               placeholder="Enter film name">
      </div>
      <!-- Director -->
```

## RESULTAT (show):

Al show mostro la info més detallada amb un poster de la peli.



## CODI (show):

Mostro la taula i el poster amb una api d'internet.

```
<div class="film-container fade-in">
    @if (!empty($film))
        <img id="backgroundPoster" src="" alt="Movie Poster Background" class="film-background">
        <div class="film-content">
            <h1 class="film-title">{{ $film['name'] }}</h1>
            <p class="film-details"><strong>Director:</strong> {{ $film['director'] }}</p>
            <p class="film-details"><strong>Any:</strong> {{ $film['year'] }}</p>
            <p class="film-details"><strong>Descripció:</strong> {{ $film['description'] }}</p>
            <a href="/films" class="back-button">Back to Films</a>
        </div>
        <img id="posterImage" src="" alt="Movie Poster" class="film-poster">
    @else
        <div class="film-content">
            <h1 class="film-title">Film Not Found</h1>
            <p class="film-details">We couldn't find the details for this film. Please return to the films page.</p>
            <a href="/films" class="back-button">Back to Films</a>
        </div>
    @endif
</div>

<script>
const movieTitle = @json($film['name'] ?? '');
const movieYear = @json($film['year'] ?? '');
const apiKey = "e897831c";
if (movieTitle && movieYear) {
    const apiUrl = `http://www.omdbapi.com/?t=${encodeURIComponent(movieTitle)}&y=${movieYear}&apikey=${apiKey}`;
    fetch(apiUrl)
        .then(response => response.json())
        .then(data => {
            if (data.Response === "True") {
                document.getElementById('backgroundPoster').src = data.Poster;
                document.getElementById('posterImage').src = data.Poster;
            } else {
                console.error("Poster not found:", data.Error);
            }
        })
        .catch(error => console.error("Error fetching poster:", error));
}
</script>
```

**RESULTAT (edit):**

Edito la peli i poso un 2 al títol i s'observa que és canvia.

## Edit Film

Film Name

Director

Year

Description

**Edit Film**

## Films

Afegir Nova Pel·lícula

Cerca per any...

Cerca per director...

| ID | TÍTOL           | DIRECTOR             | ANY  | ACCIONS |
|----|-----------------|----------------------|------|---------|
| 1  | Inception       | Christopher Nolan    | 2010 |         |
| 2  | The Godfather   | Francis Ford Coppola | 1972 |         |
| 3  | Parasite        | Bong Joon Ho         | 2019 |         |
| 4  | The Dark Knight | Christopher Nolan    | 2008 |         |
| 5  | Pulp Fiction    | Quentin Tarantino    | 1994 |         |
| 6  | Inside Out 2    | Peter Docter         | 2015 |         |

## CODI (edit):

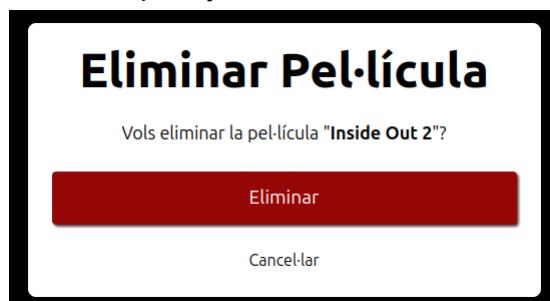
```
<!-- Edit Form -->
<form action="{{ route('films.update', $film->id) }}" method="POST">
    @csrf
    @method('PUT') <!-- Laravel PUT method for updating -->

    <!-- Film Name -->
    <div class="mb-4">
        <label for="name" class="block font-semibold">Film Name</label>
        <input type="text" id="name" name="name" value="{{ old('name', $film->name) }}" required
               class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
               placeholder="Enter film name">
    </div>

    <!-- Director -->
    <div class="mb-4">
        <label for="director" class="block font-semibold">Director</label>
        <input type="text" id="director" name="director" value="{{ old('director', $film->director) }}" required
               class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
               placeholder="Enter director name">
    </div>
```

## RESULTAT (delete):

Elimino la peli i ja no surt.



| ID | TÍTOL           | DIRECTOR             | ANY  | ACCIONS |
|----|-----------------|----------------------|------|---------|
| 1  | Inception       | Christopher Nolan    | 2010 |         |
| 2  | The Godfather   | Francis Ford Coppola | 1972 |         |
| 3  | Parasite        | Bong Joon Ho         | 2019 |         |
| 4  | The Dark Knight | Christopher Nolan    | 2008 |         |
| 5  | Pulp Fiction    | Quentin Tarantino    | 1994 |         |

## CODI (delete):

El codi de delete és simple ja que crido destroy i el filtro per id.

```
<div class="min-h-screen flex items-start justify-center fade-in pt-16 pb-16 px-4 sm:px-8 md:px-16">
  <div class="max-w-lg w-full shadow-lg rounded-lg p-6 bg-white text-black mx-auto mt-8">
    <h1 class="text-3xl sm:text-4xl md:text-5xl text-black font-extrabold mb-6 text-center">Eliminar Pel·lícula</h1>
    <p class="text-base sm:text-lg text-black mb-6 text-center">
      Vols eliminar la pel·lícula "<strong>{{ $film->name }}</strong>"?
    </p>
    <form action="{{ route('films.destroy', $film->id) }}" method="POST" class="text-center">
      @csrf
      @method('DELETE')
      <button type="submit" class="bg-dark-red text-white px-6 py-3 rounded hover:bg-black hover-animate w-full">
        Eliminar
      </button>
    </form>
    <div class="text-center mt-6">
      <a href="{{ route('films.index') }}" class="text-black hover:underline">Cancelar</a>
    </div>
  </div>
</div>
```

## RESULTAT (filtres):

Puc aplicar un filtre per any i quedaria així.

| Films                                  |                 |                   |      |         |
|--|-----------------|-------------------|------|---------|
| <a href="#">Afegir Nova Pel·lícula</a> |                 |                   |      |         |
| ID                                     | TÍTOL           | DIRECTOR          | ANY  | ACCIONS |
| 1                                      | Inception       | Christopher Nolan | 2010 |         |
| 3                                      | Parasite        | Bong Joon Ho      | 2019 |         |
| 4                                      | The Dark Knight | Christopher Nolan | 2008 |         |

O també de id directe en ordre descendent.

| ID | TÍTOL           | DIRECTOR          | ANY  | ACCIONS |
|----|-----------------|-------------------|------|---------|
| 4  | The Dark Knight | Christopher Nolan | 2008 |         |
| 3  | Parasite        | Bong Joon Ho      | 2019 |         |
| 1  | Inception       | Christopher Nolan | 2010 |         |

## CODI (filtres):

Aquest és el codi js per els filters.

```
const searchYearInput = document.getElementById('searchYear'); // Filtrar per any
const searchDirectorInput = document.getElementById('searchDirector'); // Filtrar per director

searchYearInput.addEventListener('input', function () {
    const filterYear = searchYearInput.value.toLowerCase();
    filterTableRows(filterYear, 'year');
});

searchDirectorInput.addEventListener('input', function () {
    const filterDirector = searchDirectorInput.value.toLowerCase();
    filterTableRows(filterDirector, 'director');
});

function filterTableRows(filterValue, column) {
    const rows = document.querySelectorAll('#filmsTable tbody tr');
    rows.forEach(row => {
        const cell = column === 'year' ? row.cells[3] : row.cells[2]; // Year or Director column
        const cellValue = cell.textContent.toLowerCase();
        row.style.display = cellValue.includes(filterValue) ? '' : 'none';
    });
}

const tbody = table.querySelector('tbody');
tbody.innerHTML = ''; // Clear existing rows
rows.forEach(row => {
    tbody.appendChild(row); // Re-add the row
});

// Toggle the sorting direction
sortDirection[colIndex] = !sortDirection[colIndex];

// Update the icon for sorting direction
const headers = document.querySelectorAll('.table-header i');
headers.forEach(icon => {
    icon.classList.remove('fa-sort-up', 'fa-sort-down');
    icon.classList.add('fa-sort');
});

const currentHeader = headers[colIndex];
currentHeader.classList.remove('fa-sort');
if (sortDirection[colIndex]) {
    currentHeader.classList.add('fa-sort-up');
} else {
    currentHeader.classList.add('fa-sort-down');
}
```

# CARS

## RESULTAT (index):

En aquesta pàgina mostra el contingut de la taula amb opcions de filtrar la info per any o individualment per camp en ordre ascendent o descendent.

També i han opcions per afegir, editar, esborrar o observar el cotxe en més detall.  
Finalment ho he fet responsive i amb un botó per tornar a la pàgina anterior.

The screenshot shows a dark-themed web application for managing cars. At the top, there's a header with the title "Cotxes" and a back arrow icon. Below the header, a red button says "Afegir Cotxe Nou". A search bar contains the placeholder "Cerca per any...". The main content is a table with the following data:

| ID | MARCA     | MODEL     | ANY  | PREU (€)   | ACCIONS |
|----|-----------|-----------|------|------------|---------|
| 1  | Toyota    | Corolla   | 2021 | 20000 €    |         |
| 2  | Ford      | Mustang   | 2020 | 35000 €    |         |
| 3  | Tesla     | Model S   | 2022 | 79999.99 € |         |
| 4  | Honda     | Civic     | 2019 | 18000.5 €  |         |
| 5  | Chevrolet | Silverado | 2023 | 45000 €    |         |

## CODI (index):

Creo el botó de tornar atras i creo la taula amb la funció que tinc en js per ordenar.

```
<div class="flex justify-between items-center mb-8">
  <h1 class="text-5xl font-extrabold title">Cotxes</h1>
  <div class="top-right">
    <a href="/" style="font-size: 36px;" class="text-white"> <i class="fa" style="font-size: 36px;">&#xf137;Afegir Cotxe Nou</a>
</div>
<div class="mt-4">
  <input type="text" id="search" placeholder="Cerca per any..." class="p-2 border border-gray-300 rounded w-full text-black" />
</div>
```

```
<thead>
<tr class="table-header text-sm uppercase leading-normal">
    <th class="py-3 px-6 text-left" onclick="sortTable(0)">ID <i class="fas fa-sort"></i></th>
    <th class="py-3 px-6 text-left" onclick="sortTable(1)">Marca <i class="fas fa-sort"></i></th>
    <th class="py-3 px-6 text-left" onclick="sortTable(2)">Model <i class="fas fa-sort"></i></th>
    <th class="py-3 px-6 text-left" onclick="sortTable(3)">Any <i class="fas fa-sort"></i></th>
    <th class="py-3 px-6 text-left" onclick="sortTable(4)">Preu (€) <i class="fas fa-sort"></i></th>
    <th class="py-3 px-6 text-center">Accions</th>
</tr>
</thead>
```

A la taula utilitzem el for each per mostrar cada cotxe amb les opcions filtrar per id per editar i borrar algun.

```
<tbody class="text-gray-600 text-sm font-light">
@forelse ($cars as $car)
    <tr class="table-row border-b border-gray-200">
        <td class="py-3 px-6">{{ $car->id }}</td>
        <td class="py-3 px-6">
            <a href="{{ route('cars.show', $car->id) }}" class="text-blue-500 hover:text-blue-700">
                {{ $car->make }}
            </a>
        </td>
        <td class="py-3 px-6">{{ $car->model }}</td>
        <td class="py-3 px-6">{{ $car->year }}</td>
        <td class="py-3 px-6">{{ $car->price }} €</td>
        <td class="py-3 px-6 text-center">
            <a href="{{ route('cars.edit', $car->id) }}" class="text-blue-500 hover:text-blue-700 mr-4">
                <i class="fas fa-edit"></i>
            </a>
            <a href="{{ route('cars.confirmDelete', $car->id) }}" class="text-red-500 hover:text-red-700">
                <i class="fas fa-trash-alt"></i>
            </a>
        </td>
    </tr>
@empty
    <tr>
        <td colspan="6" class="py-3 px-6 text-center">No hi ha cotxes disponibles.</td>
    </tr>
@endforelse
</tbody>
```

## RESULTAT (create):

Utilitzant un placeholder puc mostrar la info necessaria dels camps.

Add New Car

Make: BMW

Model: X5

Year: 2020

Price (€): 80000

Description: test

Save Car

Una vegada creada és veu creat i afegit a la taula

## Cotxes

Afegir Cotxe Nou

Cerca per any...

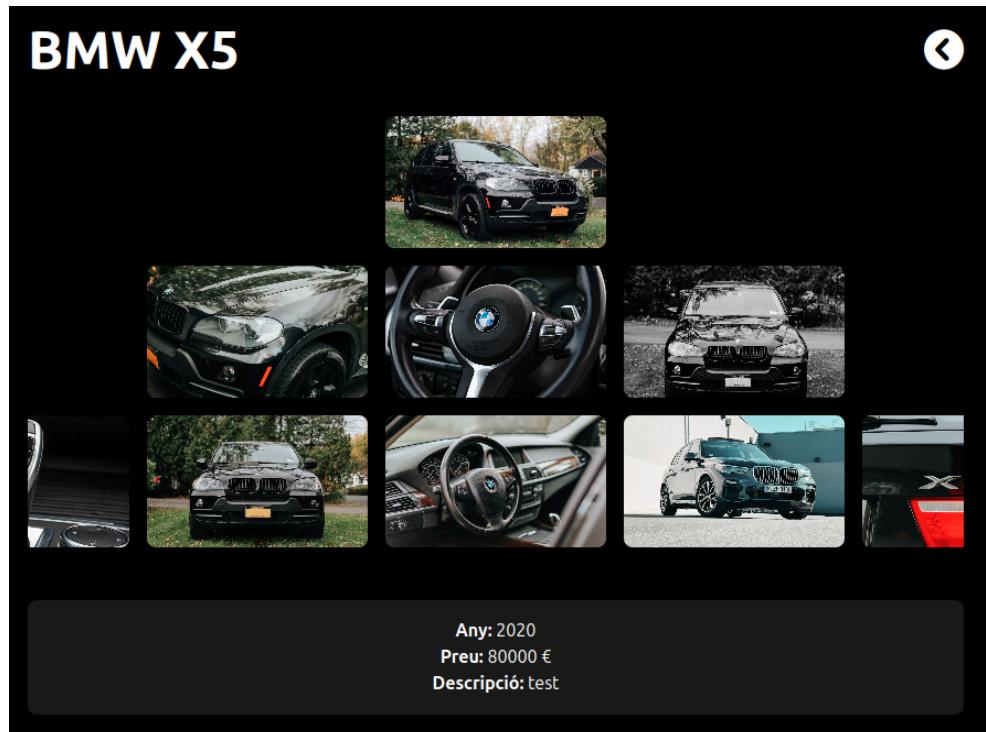
| ID | MARCA     | MODEL     | ANY  | PREU (€)   | ACCIONS |
|----|-----------|-----------|------|------------|---------|
| 1  | Toyota    | Corolla   | 2021 | 20000 €    |         |
| 2  | Ford      | Mustang   | 2020 | 35000 €    |         |
| 3  | Tesla     | Model S   | 2022 | 79999.99 € |         |
| 4  | Honda     | Civic     | 2019 | 18000.5 €  |         |
| 5  | Chevrolet | Silverado | 2023 | 45000 €    |         |
| 6  | BMW       | X5        | 2020 | 80000 €    |         |

## CODI (create):

```
<form action="{{ route('cars.store') }}" method="POST">
    @csrf
    <!-- Make -->
    <div class="mb-4">
        <label for="make" class="block font-semibold">Make</label>
        <input type="text" id="make" name="make" required
            class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
            placeholder="Enter car make">
    </div>
    <!-- Model -->
    <div class="mb-4">
        <label for="model" class="block font-semibold">Model</label>
        <input type="text" id="model" name="model" required
            class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
            placeholder="Enter model name">
    </div>
    <!-- Year -->
    <div class="mb-4">
        <label for="year" class="block font-semibold">Year</label>
```

## RESULTAT (show):

Aquí és veu el títol que serà la marca i model del cotxe, i imatges en forma de piràmide que mostra el cotxe. Per baix hi haurà info adicional.



## CODI (show):

Filtre per id i mostro el cotxe que pertany de imatges d'una api d'internet. També mostrant info del cotxe amb la taula.

```
<div class="max-w-4xl w-full mx-auto container shadow-lg rounded-lg p-6 fade-in flex-grow mt-8">
    <div class="flex justify-between items-center mb-4">
        <h1 class="text-5xl font-extrabold title">
            {{ $car ? $car->make . ' ' . $car->model : 'Cotxes' }}
        </h1>
        <div class="top-right">
            <a href="{{ route('cars.index') }}" style="font-size: 36px;" class="text-white">
                <i class="fa" style="font-size: 36px;">&#xf137;

```

**RESULTAT (edit):**

Edito el cotxe i poso un 2 al nom.

**Edit Car**

Make: BMW 2

Model: X5

Year: 2020

Price (€): 80000

Edit Car

Aquí és veu canviat.

**Cotxes**

Afegir Cotxe Nou

Cerca per any...

| ID | MARCA     | MODEL     | ANY  | PREU (€)   | ACCIONS |
|----|-----------|-----------|------|------------|---------|
| 1  | Toyota    | Corolla   | 2021 | 20000 €    |         |
| 2  | Ford      | Mustang   | 2020 | 35000 €    |         |
| 3  | Tesla     | Model S   | 2022 | 79999.99 € |         |
| 4  | Honda     | Civic     | 2019 | 18000.5 €  |         |
| 5  | Chevrolet | Silverado | 2023 | 45000 €    |         |
| 6  | BMW 2     | X5        | 2020 | 80000 €    |         |

## CODI (edit):

```
<!-- Edit Form -->
<form action="{{ route('cars.update', $car->id) }}" method="POST">
    @csrf
    @method('PUT') <!-- Laravel PUT method for updating -->

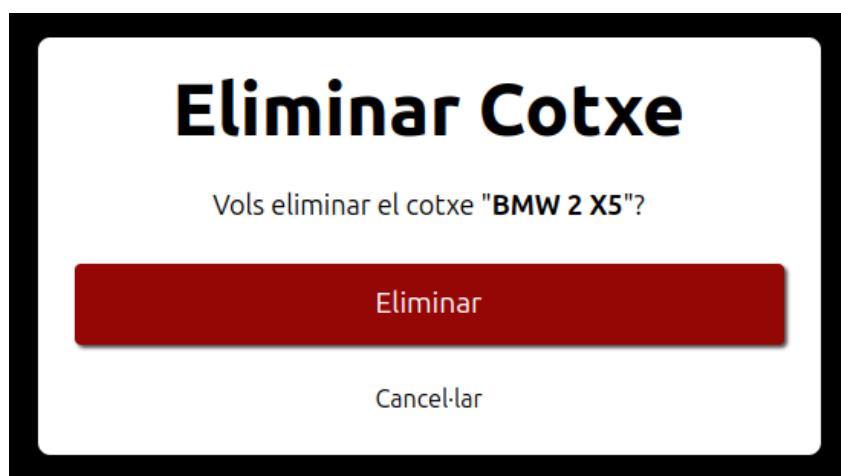
    <!-- Make -->
    <div class="mb-4">
        <label for="make" class="block font-semibold">Make</label>
        <input type="text" id="make" name="make" value="{{ old('make', $car->make) }}" required
               class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
               placeholder="Enter car make">
    </div>

    <!-- Model -->
    <div class="mb-4">
        <label for="model" class="block font-semibold">Model</label>
        <input type="text" id="model" name="model" value="{{ old('model', $car->model) }}" required
               class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
               placeholder="Enter car model">
    </div>

    <!-- Year -->
    <div class="mb-4">
        <label for="year" class="block font-semibold">Year</label>
        <input type="number" id="year" name="year" value="{{ old('year', $car->year) }}" required
               class="mt-1 block w-full border border-gray-300 rounded-md p-3 text-black"
               placeholder="Enter car year">
    </div>
```

## RESULTAT (delete):

Elimino el cotxe



Després de eliminar queda la taula així.

| ID | MARCA     | MODEL     | ANY  | PREU (€)   | ACCIONS |
|----|-----------|-----------|------|------------|---------|
| 1  | Toyota    | Corolla   | 2021 | 20000 €    |         |
| 2  | Ford      | Mustang   | 2020 | 35000 €    |         |
| 3  | Tesla     | Model S   | 2022 | 79999.99 € |         |
| 4  | Honda     | Civic     | 2019 | 18000.5 €  |         |
| 5  | Chevrolet | Silverado | 2023 | 45000 €    |         |

## CODI (delete):

Per borrar filtro per id i uso el destroy per borrar a la bd.

```
<div class="min-h-screen flex items-start justify-center fade-in pt-16 pb-16 px-4 sm:px-8 md:px-16">
  <div class="max-w-lg w-full shadow-lg rounded-lg p-6 bg-white text-black mx-auto mt-8">
    <h1 class="text-5xl text-black font-extrabold mb-6 text-center">Eliminar Cotxe</h1>
    <p class="text-lg text-black mb-6 text-center">
      Vols eliminar el cotxe "<strong>{{ $car->make }} {{ $car->model }}</strong>"?
    </p>
    <form action="{{ route('cars.destroy', $car->id) }}" method="POST" class="text-center">
      @csrf
      @method('DELETE')
      <button type="submit" class="bg-dark-red text-white px-6 py-3 rounded hover:bg-black hover-animate w-full">
        Eliminar
      </button>
    </form>
    <div class="text-center mt-6">
      <a href="{{ route('cars.index') }}" class="text-black hover:underline">Cancelar</a>
    </div>
  </div>
</div>
```

## RESULTAT (filtres):

Puc aplicar un filtre per any i quedaria així.

### Cotxes

Afegir Cotxe Nou

202

| ID | MARCA     | MODEL     | ANY  | PREU (€)   | ACCIONS |
|----|-----------|-----------|------|------------|---------|
| 1  | Toyota    | Corolla   | 2021 | 20000 €    |         |
| 2  | Ford      | Mustang   | 2020 | 35000 €    |         |
| 3  | Tesla     | Model S   | 2022 | 79999.99 € |         |
| 5  | Chevrolet | Silverado | 2023 | 45000 €    |         |

O també de id directe en ordre descendant.

### Cotxes

Afegir Cotxe Nou

202

| ID | MARCA     | MODEL     | ANY  | PREU (€)   | ACCIONS |
|----|-----------|-----------|------|------------|---------|
| 5  | Chevrolet | Silverado | 2023 | 45000 €    |         |
| 3  | Tesla     | Model S   | 2022 | 79999.99 € |         |
| 2  | Ford      | Mustang   | 2020 | 35000 €    |         |
| 1  | Toyota    | Corolla   | 2021 | 20000 €    |         |

## CODI (filtres):

Aquest és el codi js per els filters.

```
<script>
    const searchInput = document.getElementById('search');
    searchInput.addEventListener('input', function () {
        const filter = searchInput.value.toLowerCase();
        const rows = document.querySelectorAll('#carsTable tbody tr');
        rows.forEach(row => {
            const yearCell = row.cells[3].textContent.toLowerCase();
            row.style.display = yearCell.includes(filter) ? '' : 'none';
        });
    });

    function sortTable(colIndex) {
        const table = document.getElementById('carsTable');
        const rows = Array.from(table.rows).slice(1);
        rows.sort((a, b) => {
            const aText = a.cells[colIndex].textContent.trim();
            const bText = b.cells[colIndex].textContent.trim();
            if (colIndex === 0 || colIndex === 3 || colIndex === 4) {
                return sortDirection[colIndex]
                    ? parseFloat(aText) - parseFloat(bText)
                    : parseFloat(bText) - parseFloat(aText);
            } else {
                return sortDirection[colIndex]
                    ? aText.localeCompare(bText)
                    : bText.localeCompare(aText);
            }
        });
        const tbody = table.querySelector('tbody');
        tbody.innerHTML = '';
        rows.forEach(row => {
            tbody.appendChild(row);
        });

        sortDirection[colIndex] = !sortDirection[colIndex];

        const headers = document.querySelectorAll('.table-header i');
        headers.forEach(icon => {
            icon.classList.remove('fa-sort-up', 'fa-sort-down');
            icon.classList.add('fa-sort');
        });

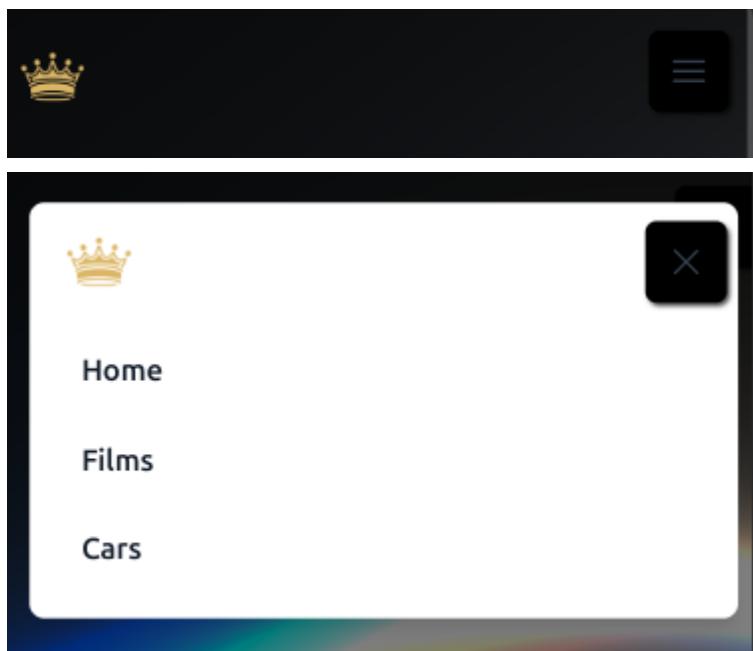
        const currentHeader = headers[colIndex];
        currentHeader.classList.remove('fa-sort');
        if (sortDirection[colIndex]) {
            currentHeader.classList.add('fa-sort-up');
        } else {
            currentHeader.classList.add('fa-sort-down');
        }
    }
}
```



# RESPONSIVITAT

## HEADER/NAVBAR:

Aquí s'observa com en mode menu el icono canvia depenent si està obert o tancat el menu.

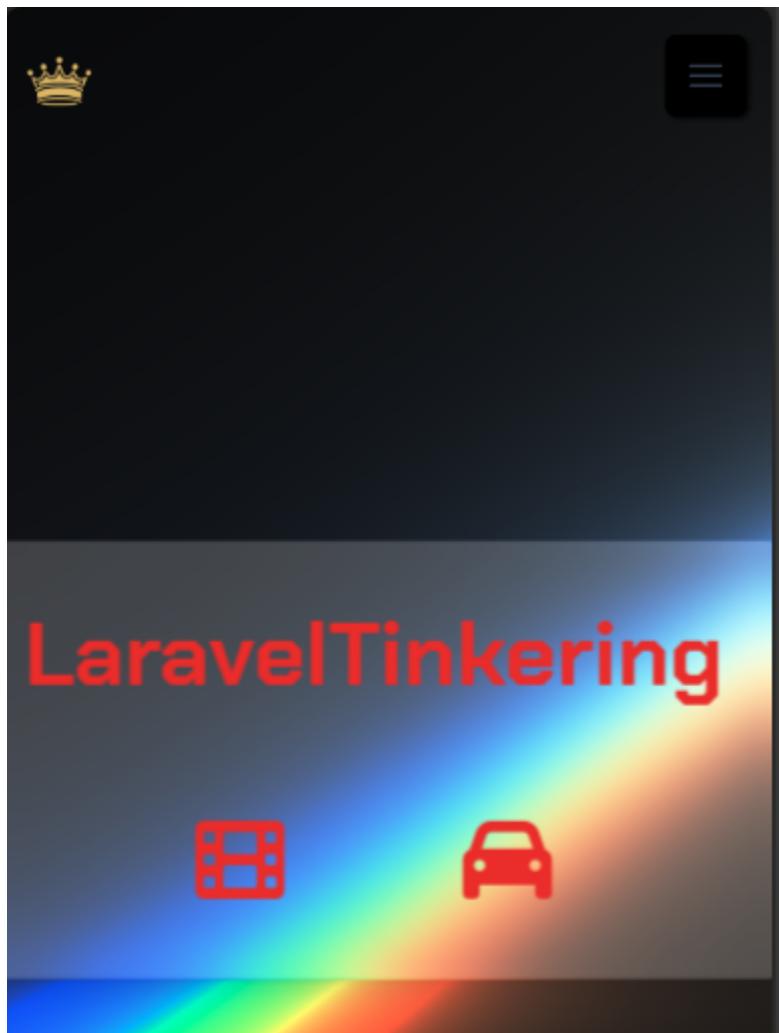


## FOOTER:

© 2024 Harvey. All Rights Reserved.



## LANDING PAGE:





## FILMS (index):

The screenshot shows a mobile application interface for managing movies. At the top, there is a navigation bar with a crown icon on the left and a menu icon on the right. Below the navigation bar, the word "Films" is displayed in large white letters. To the right of "Films" is a back arrow icon. A red button labeled "Afegir Nova Pel·lícula" is located above a search bar with the placeholder "Cerca per any...". Below the search bar is another search bar with the placeholder "Cerca per director...". The main content area features a table with movie data:

| ID | TÍTOL           | DIRECTOR             | ANY  |
|----|-----------------|----------------------|------|
| 1  | Inception       | Christopher Nolan    | 2010 |
| 2  | The Godfather   | Francis Ford Coppola | 1972 |
| 3  | Parasite        | Bong Joon Ho         | 2019 |
| 4  | The Dark Knight | Christopher Nolan    | 2008 |
| 5  | Pulp Fiction    | Quentin Tarantino    | 1994 |

At the bottom of the screen, there is a footer note: "© 2024 Harvey. All Rights Reserved."



## FILMS (create):

The image shows a mobile application interface for adding a new film. The screen has a dark background with white text and input fields. At the top, there is a navigation bar with a crown icon on the left and a menu icon on the right. Below the navigation bar, the title "Add New Film" is displayed in large red font, with a back arrow icon to its right. The form consists of several input fields: "Film Name" with placeholder "Enter film name", "Director" with placeholder "Enter director name", "Year" with placeholder "Enter release year", and "Description" with placeholder "Enter a short description (max 255 characters)". A large red button at the bottom is labeled "Save Film". At the bottom of the screen, there is a copyright notice: "© 2024 Harvey. All Rights Reserved."

Add New Film

Film Name

Enter film name

Director

Enter director name

Year

Enter release year

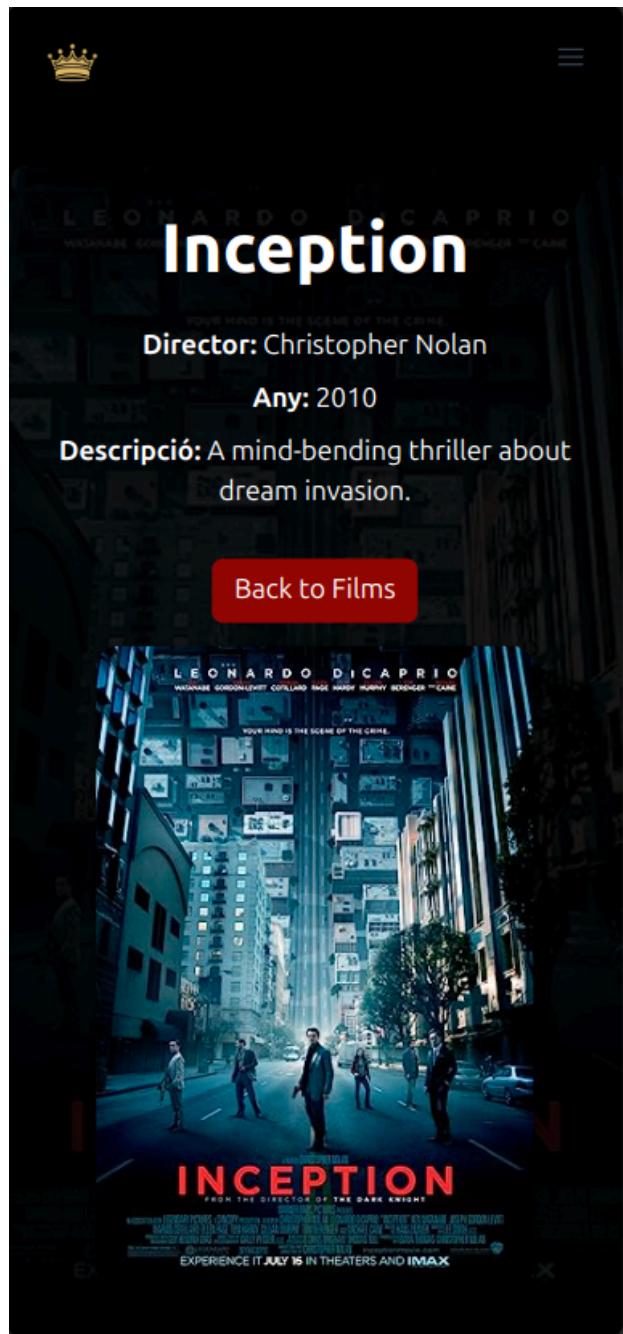
Description

Enter a short description (max 255 characters)

Save Film

© 2024 Harvey. All Rights Reserved.

## FILMS (show):





## FILMS (edit):

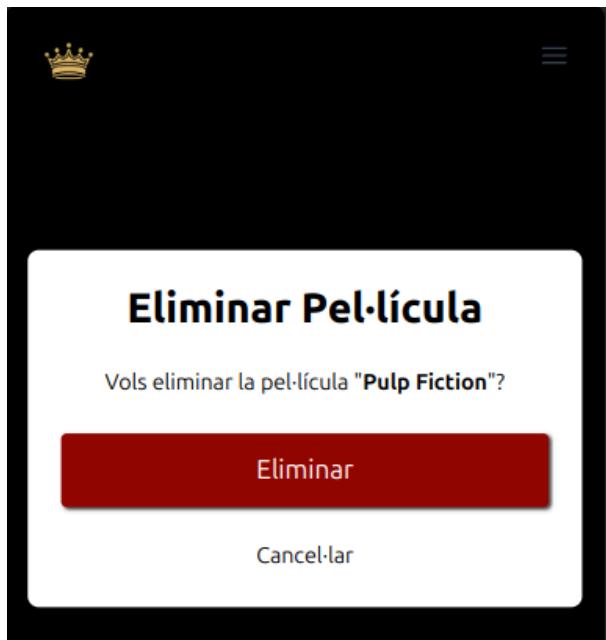
The screenshot shows a mobile application interface for editing film details. The title "Edit Film" is at the top left, and a back arrow icon is at the top right. The form fields are as follows:

- Film Name:** Inception
- Director:** Christopher Nolan
- Year:** 2010
- Description:** A mind-bending thriller about dream invasion.

A red "Edit Film" button is at the bottom of the form.



## FILMS (delete):





## CARS (index):

The screenshot shows a mobile application interface for managing a fleet of vehicles. At the top, there is a navigation bar with a crown icon on the left and a menu icon on the right. Below the navigation bar, the title "Cotxes" is displayed in large white font, with a back arrow icon to its right. A red button labeled "Afegir Cotxe Nou" is positioned below the title. A search bar with the placeholder "Cerca per any..." is also present. The main content area is a table listing five cars, each with a unique ID, brand, model, and year. The table has columns for ID, MARCA, MODEL, and ANY. The data is as follows:

| ID | MARCA     | MODEL     | ANY  |
|----|-----------|-----------|------|
| 1  | Toyota    | Corolla   | 2021 |
| 2  | Ford      | Mustang   | 2020 |
| 3  | Tesla     | Model S   | 2022 |
| 4  | Honda     | Civic     | 2019 |
| 5  | Chevrolet | Silverado | 2023 |

At the bottom of the screen, there is a copyright notice: "© 2024 Harvey. All Rights Reserved."



CARS (create):

Add New Car

Make

Enter car make

Model

Enter model name

Year

Enter manufacturing year

Price (€)

Enter car price

Description

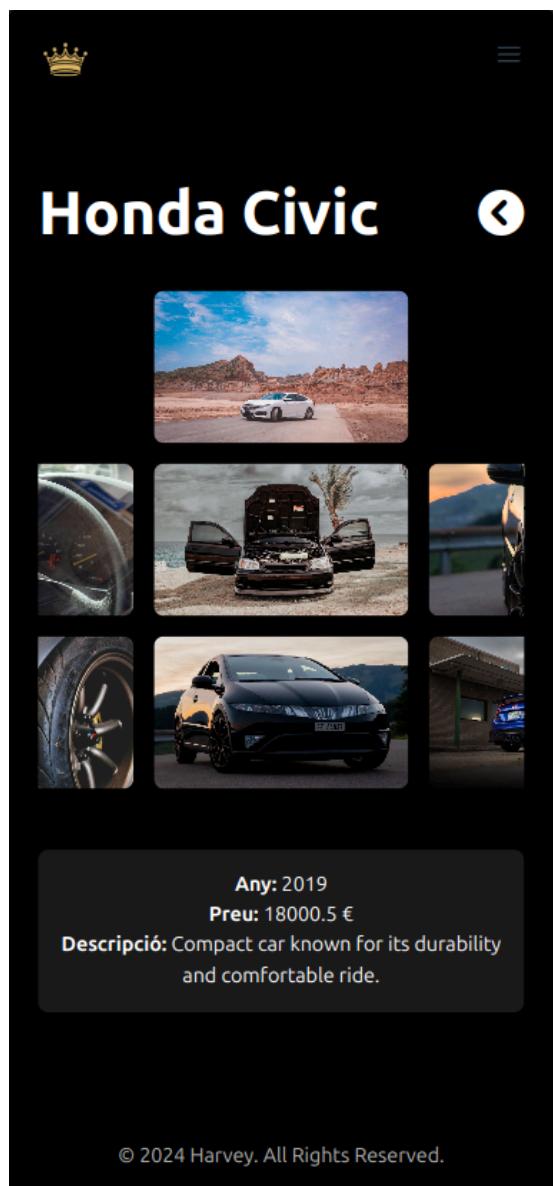
Enter a short description (max 255 characters)

Save Car

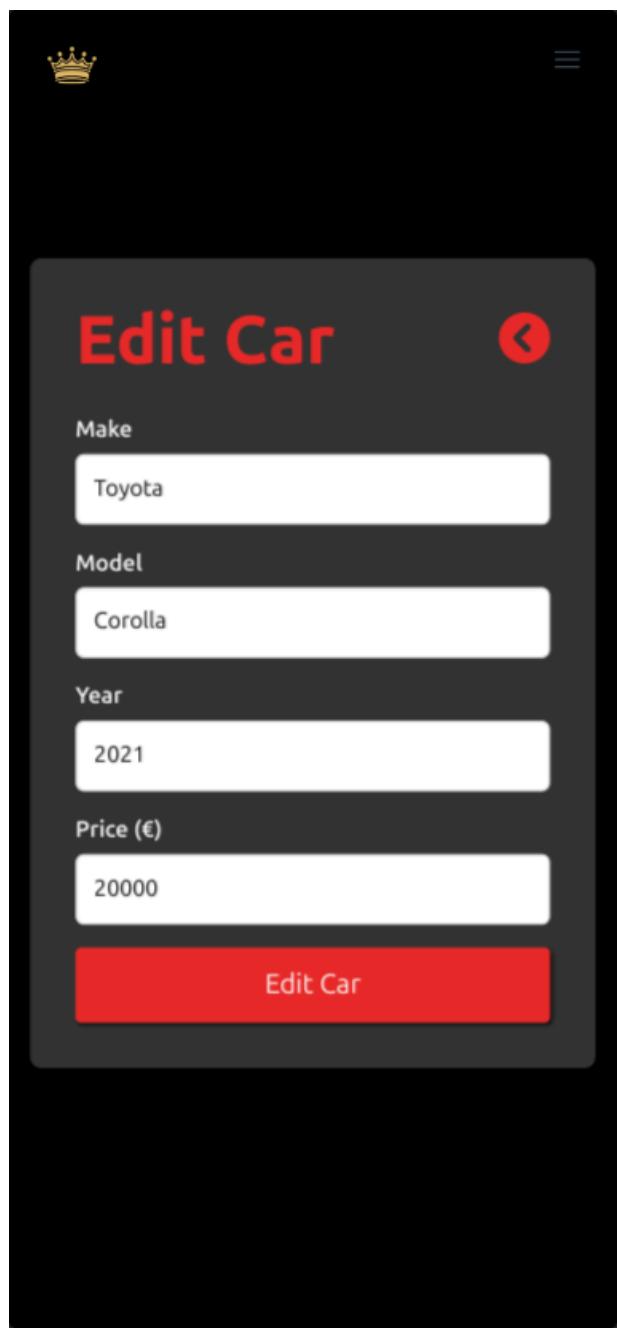
© 2024 Harvey. All Rights Reserved.



CARS (show):



## CARS (edit):





CARS (delete):

