

FRUTERIA

GUÍA PRÁCTICA: FRUTERÍA EN LARAVEL

1. Crear el proyecto

1. En la terminal, posicionarte donde guardarás el proyecto y ejecutar:

```
composer create-project laravel/laravel fruteria
```

2. En el archivo `.env`, configura la **BD**:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=fruteria
DB_USERNAME=root
DB_PASSWORD=
```

3. Crea la base de datos `fruteria` (p. ej. en phpMyAdmin).

2. Migraciones (Tablas)

Se necesitan:

1. **users**
(id, nombre, email, password, rol, timestamps)
2. **frutas**
(id, nombre, precio, temporada_id, origen_id, timestamps)
3. **temporadas**
(id, temporada, timestamps)
4. **origenes**
(id, origen, timestamps)

Ejemplo de migración para `frutas`

```
php artisan make:migration crear_tabla_frutas --create=frutas
```

```
public function up()
{
    Schema::create('frutas', function (Blueprint $table) {
        $table->id();
        $table->string('nombre');
        $table->decimal('precio', 5, 2);
        $table->unsignedBigInteger('temporada_id')->nullable();
        $table->unsignedBigInteger('origen_id')->nullable();
        $table->foreign('temporada_id')->references('id')->on('temporadas');
        $table->foreign('origen_id')->references('id')->on('origenes');
        $table->timestamps();
    });
}
```

Tras crear todas las migraciones, ejecuta:

```
php artisan migrate
```

3. Modelos

En `app/Models/`:

3.1. **User** (Por defecto)

```
namespace App\Models;

use Illuminate\Foundation\Auth\User as Authenticatable;

class User extends Authenticatable
{
    protected $table = 'users';
    protected $fillable = ['nombre','email','password','rol'];
}
```

3.2. Fruta

```
namespace App\Models;

use Illuminate\Database\Eloquent\Model;

class Fruta extends Model
{
    protected $table = 'frutas';
    protected $fillable = ['nombre','precio','temporada_id','origen_id'];

    public function temporada()
    {
        return $this->belongsTo(Temporada::class);
    }
    public function origen()
    {
        return $this->belongsTo(Origen::class);
    }
}
```

3.3. Temporada

```
namespace App\Models;

use Illuminate\Database\Eloquent\Model;

class Temporada extends Model
{
    protected $table = 'temporadas';
    protected $fillable = ['temporada'];
}
```

3.4. Origen

```
namespace App\Models;
```

```
use Illuminate\Database\Eloquent\Model;

class Origen extends Model
{
    protected $table = 'origenes';
    protected $fillable = ['origen'];
}
```

4. Seeders

Para insertar datos iniciales:

4.1. Temporadas

```
php artisan make:seeder TemporadasSeeder
```

```
use App\Models\Temporada;

class TemporadasSeeder extends Seeder
{
    public function run()
    {
        $valores = ['INVIERNO','PRIMAVERA','VERANO','OTOÑO','TODAS'];
        foreach($valores as $temp) {
            Temporada::create(['temporada' => $temp]);
        }
    }
}
```

4.2. Usuarios

```
php artisan make:seeder UsersSeeder
```

```
use App\Models\User;

class UsersSeeder extends Seeder
```

```
{
    public function run()
    {
        // Ejemplo: usuario admin
        User::create([
            'nombre' => 'Admin',
            'email' => 'admin@frutas.com',
            'rol' => 'admin',
            'password' => bcrypt('admin') // Contraseña "admin"
        ]);
    }
}
```

En `DatabaseSeeder`, registra y ejecuta:

```
public function run()
{
    $this->call([
        UsersSeeder::class,
        TemporadasSeeder::class,
        // etc.
    ]);
}
```

```
php artisan db:seed
```

o

```
php artisan migrate:fresh --seed
```

5. Autenticación con sesiones

5.1. `config/auth.php`

Verifica que `'providers' => ['users' => ['model' => App\Models\User::class]]` para usar Eloquent con el modelo `User`.

5.2. Controlador de Login

```
php artisan make:controller LoginController
```

```
namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;

class LoginController extends Controller
{
    public function loginForm()
    {
        return view('auth.login');
    }

    public function login(Request $request)
    {
        $credenciales = $request->only('login','password');
        if (Auth::attempt($credenciales)) {
            // Autenticado correctamente
            return redirect()->intended(route('frutas.index'));
        } else {
            $error = 'Credenciales incorrectas.';
            return view('auth.login', compact('error'));
        }
    }

    public function logout()
    {
        Auth::logout();
        return redirect()->route('login');
    }
}
```

Usuario y contraseña de ejemplo:

- Email (login): `admin@frutas.com`

- Password: `admin`

6. Rutas

En `routes/web.php` :

```
use App\Http\Controllers\LoginController;
use App\Http\Controllers\FrutaController;
use App\Http\Controllers\OrigenController;

// Rutas de login
Route::get('login',[LoginController::class,'loginForm'])→name('login');
Route::post('login',[LoginController::class,'login']);
Route::get('logout',[LoginController::class,'logout'])→name('logout');

// Ruta raíz → redirige a index de frutas
Route::get('/', function() {
    return redirect()→route('frutas.index');
});

// Resource para frutas (CRUD + paginación)
Route::resource('frutas', FrutaController::class);

// Ruta fruta más cara
Route::get('frutas/cara',[FrutaController::class,'masCara'])→name('frutas.c
ara');
// Ruta fruta más barata
Route::get('frutas/barata',[FrutaController::class,'masBarata'])→name('fruta
s.barata');

// Listar orígenes
Route::get('origenes',[OrigenController::class,'index'])→name('origenes.ind
ex');
```

7. Controlador `FrutaController` (Ejemplo)

```

namespace App\Http\Controllers;

use App\Models\Fruta;
use App\Models\Temporada;
use App\Models\Origen;
use Illuminate\Http\Request;

class FrutaController extends Controller
{
    // Podrías usar middleware de auth para proteger métodos
    public function __construct()
    {
        $this->middleware('auth')->except(['index','show','masCara','masBarata']);
    }

    // Lista frutas, paginado, filtrado por temporada
    public function index(Request $request)
    {
        $filtroTemp = $request->get('temporada_id');
        if ($filtroTemp) {
            $frutas = Fruta::where('temporada_id',$filtroTemp)->paginate(5);
        } else {
            $frutas = Fruta::paginate(5);
        }
        $temporadas = Temporada::all();
        return view('frutas.index', compact('frutas','temporadas'));
    }

    public function create()
    {
        $temporadas = Temporada::all();
        $origenes = Origen::all();
        return view('frutas.create', compact('temporadas','origenes'));
    }

    public function store(Request $request)
    {

```



```

$request→validate([
    'nombre' ⇒ 'required|min:2',
    'precio' ⇒ 'required|numeric|min:0',
    'temporada_id' ⇒ 'required'
]);

// Manejo del origen
if($request→filled('nuevo_origen')){
    $org = Origen::create(['origen'⇒$request→get('nuevo_origen')]);
    $origen_id = $org→id;
} else {
    $origen_id = $request→get('origen_id');
}

Fruta::create([
    'nombre' ⇒ $request→get('nombre'),
    'precio' ⇒ $request→get('precio'),
    'temporada_id' ⇒ $request→get('temporada_id'),
    'origen_id' ⇒ $origen_id
]);
return redirect()→route('frutas.index');
}

public function show($id)
{
    $fruta = Fruta::findOrFail($id);
    return view('frutas.show', compact('fruta'));
}

public function edit($id)
{
    $fruta = Fruta::findOrFail($id);
    $temporadas = Temporada::all();
    $origenes = Origen::all();
    return view('frutas.edit', compact('fruta','temporadas','origenes'));
}

public function update(Request $request, $id)

```

```

{
  $request->validate([
    'nombre' => 'required|min:2',
    'precio' => 'required|numeric|min:0'
  ]);

  $fruta = Fruta::findOrFail($id);
  $fruta->nombre = $request->get('nombre');
  $fruta->precio = $request->get('precio');
  $fruta->temporada_id = $request->get('temporada_id');

  if($request->filled('nuevo_origen')){
    $org = Origen::create(['origen'=>$request->get('nuevo_origen')]);
    $fruta->origen_id = $org->id;
  } else {
    $fruta->origen_id = $request->get('origen_id');
  }
  $fruta->save();

  return redirect()->route('frutas.index');
}

public function destroy($id)
{
  $fruta = Fruta::findOrFail($id);
  $fruta->delete();
  return redirect()->route('frutas.index');
}

// Fruta más cara
public function masCara()
{
  $fruta = Fruta::orderBy('precio','DESC')->first();
  return view('frutas.show', compact('fruta'));
}

// Fruta más barata
public function masBarata()

```

```

{
    $fruta = Fruta::orderBy('precio','ASC')→first();
    return view('frutas.show', compact('fruta'));
}
}

```

8. Controlador **OrigenController** (Listado)

```

namespace App\Http\Controllers;

use App\Models\Origen;

class OrigenController extends Controller
{
    public function index()
    {
        $origenes = Origen::all();
        return view('origenes.index', compact('origenes'));
    }
}

```

9. Vistas Blade

9.1. Plantilla principal: **layouts/plantilla.blade.php**

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>@yield('titulo')</title>
</head>
<body>
    <nav>
        @if(auth()→check())
            <a href="{{ route('frutas.index') }}">Listado Frutas</a> |
            <a href="{{ route('logout') }}">Cerrar Sesión</a>

```

```

    @else
        <a href="{{ route('login') }}">Login</a>
    @endif
</nav>
<hr>
@yield('contenido')
</body>
</html>

```

9.2. auth/login.blade.php

```

@extends('layouts.plantilla')
@section('titulo','Login')

@section('contenido')
<h1>Acceder</h1>

@if(!empty($error))
<p style="color:red">{{ $error }}</p>
@endif

<form action="{{ route('login') }}" method="POST">
    @csrf
    <label>Usuario (email):</label>
    <input type="text" name="login">
    <br>
    <label>Password:</label>
    <input type="password" name="password">
    <br><br>
    <button type="submit">Entrar</button>
</form>
@endsection

```

9.3. frutas/index.blade.php

```

@extends('layouts.plantilla')
@section('titulo','Listado de Frutas')

```

```

@section('contenido')
<h1>Listado de Frutas</h1>

<form method="GET" action="{{ route('frutas.index') }}">
  <label>Filtrar por temporada:</label>
  <select name="temporada_id">
    <option value="">-- Todas --</option>
    @foreach($temporadas as $t)
      <option value="{{ $t->id }}">{{ $t->temporada }}</option>
    @endforeach
  </select>
  <button type="submit">Filtrar</button>
</form>

@if(auth()->check())
  <a href="{{ route('frutas.create') }}">Nueva Fruta</a>
@endif

<ul>
@forelse($frutas as $fruta)
  <li>
    <a href="{{ route('frutas.show',$fruta->id) }}">
      {{ $fruta->nombre }} ({{ $fruta->precio }} €)
    </a>
    @if(auth()->check())
      <a href="{{ route('frutas.edit',$fruta->id) }}">Editar</a>
      <form action="{{ route('frutas.destroy',$fruta->id) }}" method="POST" style="display:inline">
        @csrf
        @method('DELETE')
        <button type="submit">Borrar</button>
      </form>
    @endif
  </li>
@empty
  <li>No hay frutas</li>
@endforelse

```

```
</ul>
```

```
{{ $frutas→links() }}
```

```
@endsection
```

9.4. frutas/create.blade.php

```
@extends('layouts.plantilla')
```

```
@section('titulo','Nueva Fruta')
```

```
@section('contenido')
```

```
<h1>Nueva Fruta</h1>
```

```
@if($errors→any())
```

```
<ul style="color:red">
```

```
    @foreach($errors→all() as $err)
```

```
        <li>{{ $err }}</li>
```

```
    @endforeach
```

```
</ul>
```

```
@endif
```

```
<form action="{{ route('frutas.store') }}" method="POST">
```

```
    @csrf
```

```
    <p>Nombre:<input type="text" name="nombre" value="{{ old('nombre') }}"></p>
```

```
    <p>Precio (€):<input type="text" name="precio" value="{{ old('precio') }}"></p>
```

```
    <p>Temporada:
```

```
    <select name="temporada_id">
```

```
        @foreach($temporadas as $temp)
```

```
            <option value="{{ $temp→id }}">{{ $temp→temporada }}</option>
```

```
        @endforeach
```

```
    </select>
```

```
</p>
```

```
<p>
```

```
Origen existente:
```

```
<select name="origen_id">
```

```
    <option value="">-- Seleccionar --</option>
```

```

    @foreach($origenes as $org)
    <option value="{{ $org->id }}">{{ $org->origen }}</option>
    @endforeach
</select>
</p>
<p>O nuevo origen: <input type="text" name="nuevo_origen"></p>
<button type="submit">Guardar</button>
</form>
@endsection

```

9.5. frutas/edit.blade.php

Similar a create, pero con datos precargados:

```

@extends('layouts.plantilla')
@section('titulo','Editar Fruta')

@section('contenido')
<h1>Editar Fruta</h1>

@if($errors->any())
<ul style="color:red">
    @foreach($errors->all() as $err)
    <li>{{ $err }}</li>
    @endforeach
</ul>
@endif

<form action="{{ route('frutas.update',$fruta->id) }}" method="POST">
    @csrf
    @method('PUT')
    <p>Nombre:<input type="text" name="nombre" value="{{ old('nombre', $fruta->nombre) }}"></p>
    <p>Precio:<input type="text" name="precio" value="{{ old('precio', $fruta->precio) }}"></p>
    <p>Temporada:
    <select name="temporada_id">
        @foreach($temporadas as $temp)

```

```

<option value="{{ $temp→id }}"
    {{ $fruta→temporada_id == $temp→id ? 'selected' : '' }}>
    {{ $temp→temporada }}
</option>
@endforeach
</select>
</p>
<p>
Origen existente:
<select name="origen_id">
    <option value="">-- Seleccionar --</option>
    @foreach($origenes as $org)
        <option value="{{ $org→id }}"
            {{ $fruta→origen_id == $org→id ? 'selected' : '' }}>
            {{ $org→origen }}
        </option>
    @endforeach
</select>
</p>
<p>O nuevo origen: <input type="text" name="nuevo_origen"></p>

<button type="submit">Actualizar</button>
</form>
@endsection

```

9.6. frutas/show.blade.php

```

@extends('layouts.plantilla')
@section('titulo','Detalle Fruta')

@section('contenido')
<h1>Detalle de Fruta</h1>

@if($fruta)
    <p><strong>Nombre:</strong> {{ $fruta→nombre }}</p>
    <p><strong>Precio:</strong> {{ $fruta→precio }} €</p>
    <p><strong>Temporada:</strong> {{ $fruta→temporada→temporada ??
'Ninguna' }}</p>

```



```

    <p><strong>Origen:</strong> {{ $fruta→origen→origen ?? 'Desconocid
o' }}</p>
@else
    <p>No se encontró la fruta solicitada.</p>
@endif

<a href="{{ route('frutas.index') }}">Volver al listado</a>
@endsection

```

9.7. `origenes/index.blade.php`

```

@extends('layouts.plantilla')
@section('titulo','Listado de Orígenes')

@section('contenido')
    <h1>Orígenes</h1>
    @if(count($origenes) > 0)
        <ul>
            @foreach($origenes as $org)
                <li>{{ $org→origen }}</li>
            @endforeach
        </ul>
    @else
        <p>No hay orígenes registrados.</p>
    @endif

    <a href="{{ route('frutas.index') }}">Volver</a>
@endsection

```