



SERVICIO WEB BAJO EL ESTILO DE ARQUITECTURA REST

Películas Service

integración de soluciones

periodo 10/agosto / 2021 – 31/enero/2022

José Fernando Guevara Espíritu

Contenido

Película.cs.....	1
Director.cs	1
PeliculaDTO.cs.....	1
PeliculaDetail.DTO.....	2
PeliculaController.cs.....	2
App.js	4
Index.html	6

Película.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Web;

namespace PeliculaService3.Models
{
    public class Pelicula
    {
        public int Id { get; set; }
        [Required]
        public string Titulo { get; set; }
        public string Genero { get; set; }
        public decimal Precio { get; set; }
        public int Anio { get; set; }

        public int DirectorId { get; set; }
        public Director Director { get; set; }
    }
}
```

Director.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Web;

namespace PeliculaService3.Models
{
    public class Director
    {
        public int Id { get; set; }
        [Required]
        public string Nombre { get; set; }
    }
}
```

PeliculaDTO.cs

```
using System;
using System.Collections.Generic;
```

```

using System.Linq;
using System.Web;

namespace PeliculaService3.Models
{
    public class PeliculaDTO
    {
        public int Id { get; set; }
        public string Titulo { get; set; }
        public string DirectorNombre { get; set; }
    }
}

```

PeliculaDetail.DTO

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace PeliculaService3.Models
{
    public class PeliculaDetailDTO
    {
        public int Id { get; set; }
        public string Titulo { get; set; }
        public string Genero { get; set; }
        public decimal Precio { get; set; }
        public string DirectorNombre { get; set; }
        public int Anio { get; set; }
    }
}

```

PeliculaController.cs

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Threading.Tasks;
using System.Web.Http;
using System.Web.Http.Description;
using PeliculaService3.Data;
using PeliculaService3.Models;

```

```

namespace PeliculaService3.Controllers
{
    public class PeliculasController : ApiController
    {
        private PeliculaService3Context db = new PeliculaService3Context();

        public IQueryable<PeliculaDTO> GetBooks()
        {
            // GET: api/peliculas

            var peliculas = from b in db.Peliculas
                            select new PeliculaDTO()
                            {
                                Id = b.Id,
                                Titulo = b.Titulo,
                                DirectorNombre = b.Director.Nombre
                            };
            return peliculas;
        }

        // GET: api/Pelicula/5
        [ResponseType(typeof(PeliculaDetailDTO))]
        public async Task<IHttpActionResult> GetBook(int id)
        {
            var pelicula = await db.Peliculas.Include(b => b.Director).Select(b =>
                new PeliculaDetailDTO()
                {
                    Id = b.Id,
                    Titulo = b.Titulo,
                    Anio = b.Anio,
                    Precio = b.Precio,
                    DirectorNombre = b.Director.Nombre,
                    Genero = b.Genero
                }).SingleOrDefaultAsync(b => b.Id == id);
            if (pelicula == null)
            {
                return NotFound();
            }
            return Ok(pelicula);
        }

        // POST: api/Books
        [ResponseType(typeof(Pelicula))]
        public async Task<IHttpActionResult> PostBook(Pelicula pelicula)
        {
            if (!ModelState.IsValid)
            {
                return BadRequest(ModelState);
            }
            db.Peliculas.Add(pelicula);
            await db.SaveChangesAsync();
            // New code:
            // Load author name
            db.Entry(pelicula).Reference(x => x.Director).Load();
            var dto = new PeliculaDTO()
            {
                Id = pelicula.Id,
            }
        }
    }
}

```

```

        Titulo = pelicula.Titulo,
        DirectorNombre = pelicula.Director.Nombre
    };
    return CreatedAtRoute("DefaultApi", new { id = pelicula.Id }, dto);
}

[ResponseType(typeof(Pelicula))]
public async Task<IHttpActionResult> DeletePelicula(int id)
{
    Pelicula pelicula = await db.Peliculas.FindAsync(id);
    if (pelicula == null)
    {
        return NotFound();
    }

    db.Peliculas.Remove(pelicula);
    await db.SaveChangesAsync();

    return Ok(pelicula);
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}

private bool PeliculaExists(int id)
{
    return db.Peliculas.Count(e => e.Id == id) > 0;
}
}

```

App.js

```

var peliculasUri = '/api/peliculas/';
var directorsUri = '/api/directors/';

function ajaxHelper(uri, method, data) {

    $('#error').html('');
    return $.ajax({
        type: method,
        url: uri,
        dataType: 'json',
        contentType: 'application/json',
        data: data ? JSON.stringify(data) : null
    }).fail(function (jqXHR, textStatus, errorThrown) {
        $('#error').html(errorThrown);
    });
}

```

```

}

function getDirectores() {
    ajaxHelper(directorsUri, 'GET').done(function (data) {
        ImprimeDirectores(data);
    });
}

function getAllPelículas() {
    ajaxHelper(películasUri, 'GET').done(function (data) {
        ImprimeLista(data);
    });
}

function ImprimeDirectores(jsondata) {
    var directors = $('#DirectorId');

    $(directors).empty();
    $.each(jsondata, function (i, obj) {
        $(directors).append($('

```

```

        Genero: $('#inputGenero').val(),
        Precio: $('#inputPrecio').val(),
        Titulo: $('#inputTitulo').val(),
        Anio: $('#inputtoAnio').val()
    });

    ajaxHelper(peliculasUri, 'POST', pelicula).done(function (item) {
        getAllPeliculas();
    });
    return false;
}

/*function eliminar()
{
    $("#borrar").on("click", function () {
        alert('estoy en onclick');
    });
}*/

// Fetch the initial data.
getAllPeliculas();
getDirectors();
//eliminar();

```

Index.html

```

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Películas REST </title>

    <script src="https://code.jquery.com/jquery-2.2.4.min.js"></script>

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" />

    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

    <script src="Scripts/app.js"></script>
</head>
<body>
    <div class="container">

        <div class="page-header">
            <h1>Netflix</h1>
        </div>

        <div class="alert alert-danger"><p id="error"></p></div>

```



```

<div class="row">

  <div class="col-md-4">
    <div class="panel panel-default">
      <div class="panel-heading">
        <h2 class="panel-titulo">Películas</h2>
      </div>
      <div class="panel-body">
        <ul id="películas" class="list-unstyled">
          <li>
            <input type="hidden" id="películaId">
            <strong>
              <span class="DirectorNombre"></span>
            </strong>:
            <span class="Titulo"></span>
            <small><a href="#">Detalles</a></small>

            <button type="button" class="btn btn-default"
id="borrar" >Borrar</button>
          </li>

        </ul>
      </div>
    </div>

  </div>

  <div class="col-md-4">
    <!-- TODO: Book details -->
    <div class="panel panel-default">
      <div class="panel-heading">
        <h2 class="panel-title">detalles</h2>
      </div>
      <table class="table table-detalles">
        <tr><td>Director</td><td class="DirectorNombre"></td></tr>
        <tr><td>Titulo</td><td class="Titulo"></td></tr>
        <tr><td>Anio</td><td class="Anio"></td></tr>
        <tr><td>Genero</td><td class="Genero"></td></tr>
        <tr><td>Precio</td><td class="Precio"></td></tr>
      </table>
    </div>

  </div>

  <div class="col-md-4">
    <!-- TODO: Add new book -->
    <div class="panel panel-default">
      <div class="panel-heading">
        <h2 class="panel-Titulo">Add Película</h2>
      </div>

      <div class="panel-body">
        <form class="form-horizontal" onsubmit="return
addPelícula();">

          <div class="form-group">
            <label for="inputAuthor" class="col-sm-2 control-
label">Director</label>

```

```

        <div class="col-sm-10">
            <select id="DirectorId" class="form-
control"></select>
        </div>
    </div>
    <div class="form-group">
        <label for="inputTitulo" class="col-sm-2 control-
label">Titulo</label>
        <div class="col-sm-10">
            <input type="text" class="form-control"
id="inputTitulo" />
        </div>
        <label for="inputAnio" class="col-sm-2 control-
label">Anio</label>
        <div class="col-sm-10">
            <input type="number" class="form-control"
id="inputAnio" />
        </div>
        <label for="inputGenero" class="col-sm-2 control-
label">Genero</label>
        <div class="col-sm-10">
            <input type="text" class="form-control"
id="inputGenero" />
        </div>
        <label for="inputPrecio" class="col-sm-2 control-
label">Precio</label>
        <div class="col-sm-10">
            <input type="number" step="any" class="form-
control" id="inputPrecio" />
        </div>
        <button type="submit" class="btn btn-
default">Agregar</button>
    </form>
</div>
</div>
</div>
</body>
</html>

```