

## Breve descripción del proyecto.

### Dirección del repositorio:

[https://github.com/aamartin2k/Museum\\_API\\_Challenge.git](https://github.com/aamartin2k/Museum_API_Challenge.git)

### Estructura:

- Servicio Web
- Elementos compartidos
- Cliente

### Targeting:

**Servicio Web:** NET Core 2.2 Multiplataforma (Windows, macOS, LINUX).

**Cliente:** Net Framework 4.6 Windows.

**Shared:** Ambas plataformas.

#### Shared.

El elemento compartido implementa tipos (clases) y constantes comunes. Permite tener tipo fuerte del lado del cliente. Permite mantener una misma base de código con proyectos que apuntan a diferentes plataformas: Net Framework y NET Core.

Establece las bases para desarrollar el SDK de la API.

#### Cliente.

El Cliente es una Aplicación Consola de Windows. Es muy simple, nada atractivo visualmente, es una demostración rápida de cómo usar la biblioteca **RestSharp** para consumir la API.

Implementa una clase cliente para cada endpoint. Todos los clientes heredan de una clase base abstracta. Y además exponen una interfase al código consumidor final, lo que libera al cliente de tener una referencia dura a las capas inferiores.

El archivo *demoLog.txt* muestra el resultado de la ejecución del cliente consumiendo la API.

#### Servicio Web.

Se estructura en tres capas: Controlador, Servicio y Repositorio. Servicio y Repositorio implementan interfaces y se inyectan las dependencias. Se reduce dependencias entre módulos y facilita desarrollo colectivo y la implementación de pruebas unitarias.

Se aplica mapping para separar los tipos empleados en el modelo de los recursos servidos a clientes. Permite incorporar hypermedia a los recursos.

En la capa Repositorio se aplica el patrón Unit of Work. Por comodidad y agilidad, se utiliza persistencia en memoria.