

Erick Francisco De la Rosa

Estructura y Carga de Dato

Documentación Técnica de la Solución C#

1. Introducción

Este documento describe la configuración e implementación inicial de la solución C# desarrollada en Visual Studio. La solución se compone de un proyecto de aplicación principal y un proyecto de pruebas unitarias. Se incluyen dependencias esenciales para la carga de datos y pruebas, como **CsvHelper** y **xUnit**.

2. Objetivo

El objetivo principal de esta configuración es establecer una base sólida para el desarrollo del sistema, permitiendo:

- La carga y procesamiento de datos desde archivos CSV y JSON.
- La realización de pruebas unitarias de forma automatizada.
- Una organización clara de la solución para facilitar la escalabilidad del proyecto.

3. Estructura del Proyecto

La solución se organiza de la siguiente manera:

```
/[Root] └─ Carga-datos.sln // Archivo de solución de Visual Studio └─ /src //
Proyecto principal de la aplicación | └─ PIBImpact.csproj // Archivo de proyecto
para la aplicación | └─ Program.cs // Punto de entrada de la aplicación | └─
/Models // Clases de modelo (ej. Pais, SectorEconomico, Arancel) | | └─ Pais.cs
| | └─ SectorEconomico.cs | | └─ Arancel.cs | └─ /Data // Clases para carga y
manejo de datos (ej. DataLoader.cs) | └─ DataLoader.cs └─ /tests // Proyecto de
pruebas unitarias | └─ PIBImpact.Tests.csproj // Archivo de proyecto de pruebas
(xUnit) | └─ DataLoaderTests.cs // Pruebas unitarias para validar la carga de datos
└─ /docs // Documentación técnica y otros recursos └─
implementacion_tecnica.md // Documento actual
```

4. Configuración de la Solución

4.1. Proyecto Principal (PIBImpact)

El proyecto principal se encuentra en el directorio ``/src`` y está configurado como una aplicación de consola utilizando el SDK de .NET. El archivo ``PIBImpact.csproj`` define las siguientes propiedades y dependencias:

- **OutputType**: Exe (aplicación de consola).
- **TargetFramework**: ``net6.0`` (puede actualizarse a la versión deseada, por ejemplo, ``net7.0`` o ``net5.0``).
- **Dependencia principal**:
 - [CsvHelper](https://joshclose.github.io/CsvHelper/), que facilita la lectura y mapeo de archivos CSV.

Ejemplo de ``PIBImpact.csproj``:

``<<<xml``

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <OutputType>Exe</OutputType>
    <TargetFramework>net6.0</TargetFramework>
  </PropertyGroup>
  <ItemGroup>
    <PackageReference Include="CsvHelper" Version="27.2.1" />
  </ItemGroup>
</Project>
```

4.2. Proyecto de Pruebas Unitarias (PIBImpact.Tests)

El proyecto de pruebas se encuentra en el directorio `/tests` y utiliza **xUnit** para la ejecución de pruebas unitarias. Este proyecto también hace referencia al proyecto principal para permitir la validación de su funcionalidad.

Ejemplo de PIBImpact.Tests.csproj:

```
<Project Sdk="Microsoft.NET.Sdk">

  <PropertyGroup>

    <TargetFramework>net6.0</TargetFramework>

    <IsPackable>>false</IsPackable>

  </PropertyGroup>

  <ItemGroup>

    <PackageReference Include="xunit" Version="2.4.1" />

    <PackageReference Include="xunit.runner.visualstudio" Version="2.4.3" />

    <PackageReference Include="Microsoft.NET.Test.Sdk" Version="17.0.0" />

  </ItemGroup>

  <ItemGroup>

    <ProjectReference Include="..\src\PIBImpact.csproj" />

  </ItemGroup>

</Project>
```

5. Dependencias Iniciales

La configuración inicial incluye las siguientes dependencias:

- **CsvHelper**: Librería utilizada para facilitar la lectura y mapeo de archivos CSV.
- **xUnit**: Framework para la ejecución de pruebas unitarias y validación del código.

Estas dependencias permiten establecer una base sólida para el desarrollo de funcionalidades adicionales y garantizar la calidad del código mediante pruebas automatizadas.