

TUTORIAL DE UNA TECNOLOGÍA/FRAMEWORK: .NET 9 WEB API

Jesús Alberto Castillo Duarte
Gonzalo Xochipilli Flores Valdes
Vicente Ulises Olivares Plasencia
Jorge Alejandro Núñez Delgado
Ysmael Paredes Gonzalez
Alexia Elizabeth Yáñez Hidalgo

UNIR

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

Contents

1. Justificación del framework ¿Para qué sirve?	2
¿Qué aplicaciones tiene?	2
¿Sobre qué lenguajes se apoya?	2
2. Instalación	2
Pasos de instalación:	3
Rutas importantes.....	3
Variables de entorno.....	3
¿Necesita base de datos?.....	3
3. Primeros pasos.....	4
Hola Mundo con .NET 9 Web API.....	4
4. Utilización del framework	5
¿Cómo empezar a crear una aplicación base?.....	5
5. Explicación del funcionamiento del framework.....	9
¿Actúa sobre páginas, ficheros, secciones...?	9
¿Cómo se editan/modifican los ficheros?	9
¿Cómo se referencian desde otras aplicaciones?	9
6. Conclusiones	9

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

Curso: Desarrollo de APIs con .NET 9

1. Justificación del framework

¿Para qué sirve?

.NET 9 es un framework de desarrollo creado por Microsoft que permite construir aplicaciones modernas, seguras y multiplataforma. Su objetivo principal es ofrecer un entorno unificado para desarrollar **aplicaciones web, APIs, microservicios, aplicaciones de escritorio, móviles y en la nube**, con un rendimiento excepcional.

¿Qué aplicaciones tiene?

- **Web APIs** y microservices (.NET Web API).
- **Aplicaciones web** con ASP.NET Core (MVC, Razor Pages, Blazor).
- **Aplicaciones móviles** (con .NET MAUI).
- **Aplicaciones de escritorio** (WPF, WinForms).
- **Procesos en la nube** (Azure Functions, AWS Lambda).
- **Juegos** (Unity usa C# y .NET).
- **Inteligencia artificial y análisis de datos** (con ML.NET).

¿Sobre qué lenguajes se apoya?

Principalmente sobre:

- **C#** (lenguaje principal, moderno, seguro y de tipado fuerte).
- **F#** (funcional, ideal para cálculos matemáticos o científicos).
- **VB.NET** (compatibilidad con sistemas legados).

2. Instalación

Sí, se requiere instalar el SDK de .NET 9 y un entorno de desarrollo compatible (Visual Studio, Visual Studio Code o JetBrains Rider).

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

Pasos de instalación:

1. Descargar el SDK de .NET 9:

- <https://dotnet.microsoft.com/download>

2. Verificar la instalación:

- dotnet --version

```
Microsoft Windows [Version 10.0.26200.7019]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chiro>dotnet --version
9.0.300

C:\Users\chiro>
```

3. Instalar un IDE:

- **Visual Studio 2022 17.9+** (Windows)
- **Visual Studio Code** (multiplataforma)
- **JetBrains Rider** (multiplataforma)

Rutas importantes

- SDK: C:\Program Files\dotnet\
- Herramientas globales: %USERPROFILE%\.dotnet\tools\
- En Linux/macOS: /usr/local/share/dotnet/

Variables de entorno

El instalador normalmente las configura automáticamente:

- DOTNET_ROOT=C:\Program Files\dotnet
- PATH=%PATH%;C:\Program Files\dotnet\

¿Necesita base de datos?

Para proyectos Web API básicos, no.

Pero si se persisten datos, se recomienda:

- **SQL Server Express o PostgreSQL**
- Acceso mediante **Entity Framework Core 9**

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

3. Primeros pasos

Hola Mundo con .NET 9 Web API

Ejecuta los siguientes comandos:

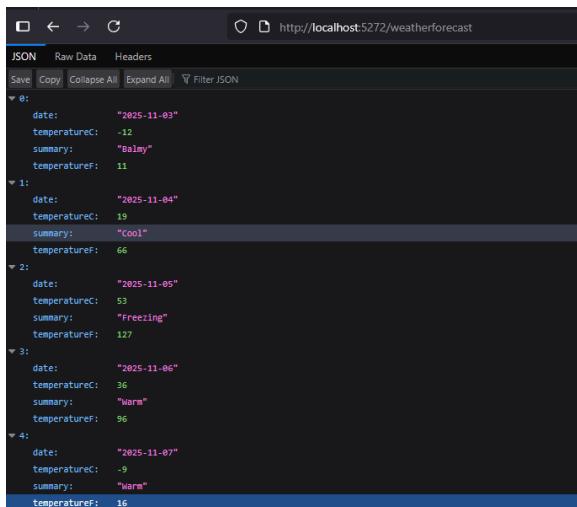
- dotnet new webapi -n HelloWorldApi
- cd HelloWorldApi
- dotnet run

```
PS C:\Users\chiro\OneDrive\Desktop\TareaTutorialFramework> dotnet new webapi -n HelloWorldApi
The template "ASP.NET Core Web API" was created successfully.

Processing post-creation actions...
Restoring C:\Users\chiro\OneDrive\Desktop\TareaTutorialFramework\HelloWorldApi\HelloWorldApi.csproj...
Restore succeeded.

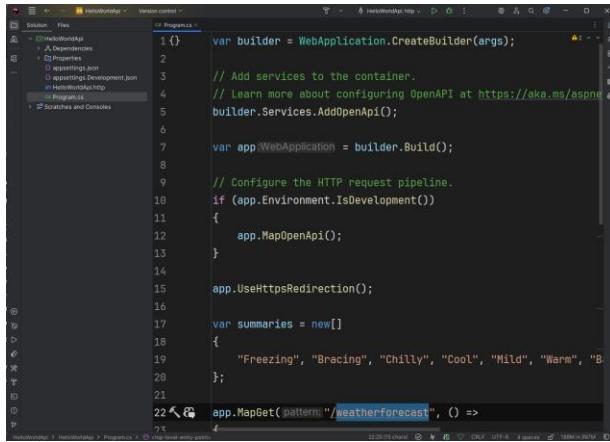
PS C:\Users\chiro\OneDrive\Desktop\TareaTutorialFramework> cd HelloWorldApi
PS C:\Users\chiro\OneDrive\Desktop\TareaTutorialFramework\HelloWorldApi> dotnet run
Using launch settings from C:\Users\chiro\OneDrive\Desktop\TareaTutorialFramework\HelloWorldApi\Properties\launchSettings.json...
Building...
[info]: Microsoft.Hosting.Lifetime[14]
  Now listening on: http://localhost:5272
[info]: Microsoft.Hosting.Lifetime[0]
  Application started. Press Ctrl+C to shut down.
[info]: Microsoft.Hosting.Lifetime[0]
  Hosting environment: Development
[info]: Microsoft.Hosting.Lifetime[0]
  Content root path: C:\Users\chiro\OneDrive\Desktop\TareaTutorialFramework\HelloWorldApi
```

Navega a <http://localhost:5272/weatherforecast>



Abre los archivos en tu IDE (en mi caso JetBrains Rider):

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025



```

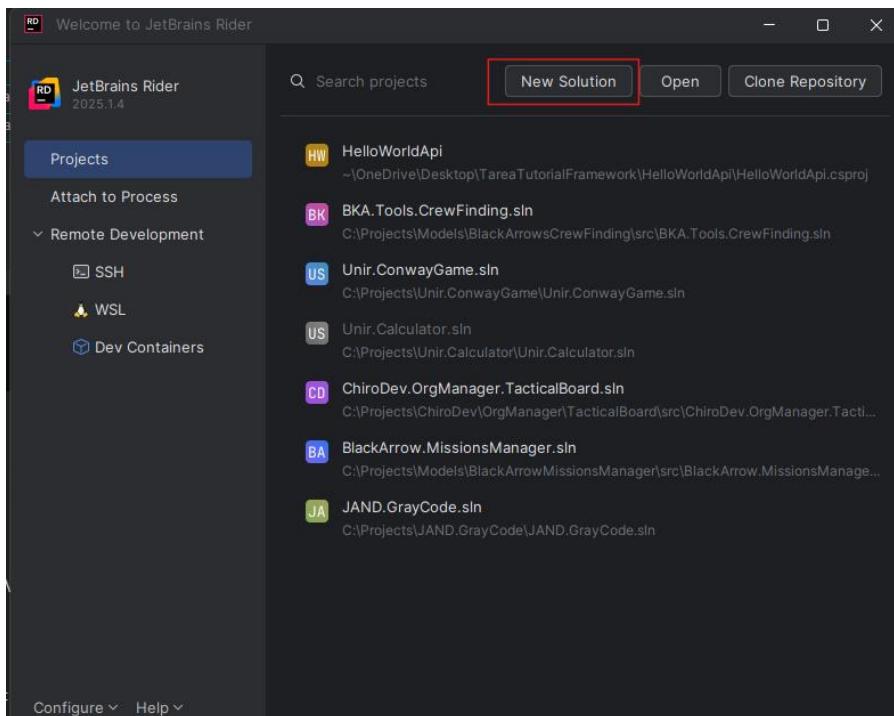
1 var builder = WebApplication.CreateBuilder(args);
2
3 // Add services to the container.
4 // Learn more about configuring OpenAPI at https://aka.ms/aspnet-openapi
5 builder.Services.AddOpenApi();
6
7 var app = builder.Build();
8
9 // Configure the HTTP request pipeline.
10 if (app.Environment.IsDevelopment())
11 {
12     app.MapOpenApi();
13 }
14
15 app.UseHttpsRedirection();
16
17 var summaries = new[]
18 {
19     "Freezing", "Bracing", "Chilly", "Cool", "Mild", "Warm", "Balmy",
20 };
21
22 app.MapGet("/weatherforecast", () =>
23     new WeatherForecast()
24     {
25         Summary = summaries[Random.Shared.Next(summaries.Length)],
26         TemperatureC = Random.Shared.Next(-20, 45),
27         TemperatureF = 32 + (9 * TemperatureC / 5)
28     }
29 );

```

4. Utilización del framework

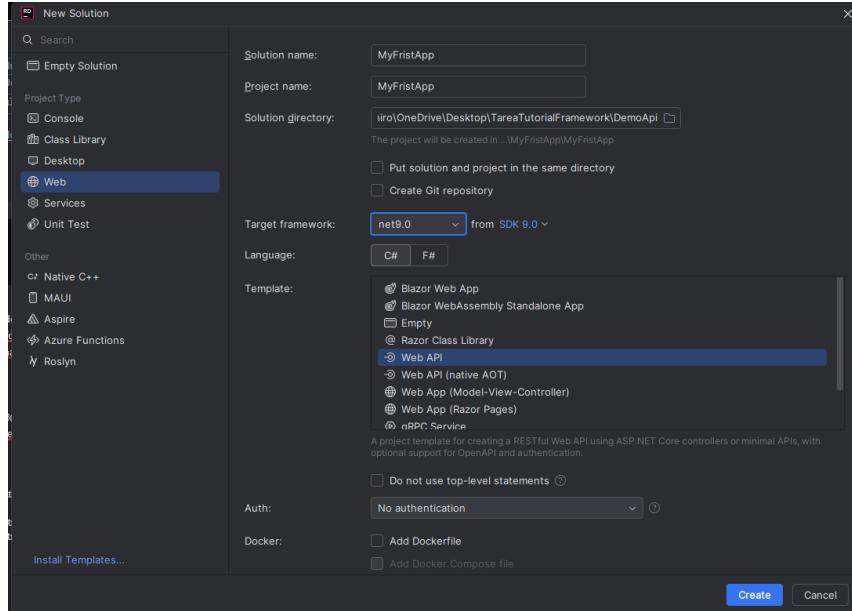
¿Cómo empezar a crear una aplicación base?

1. Crea un nuevo Proyecto en tu IDE (JetBrains Rider en mi caso):

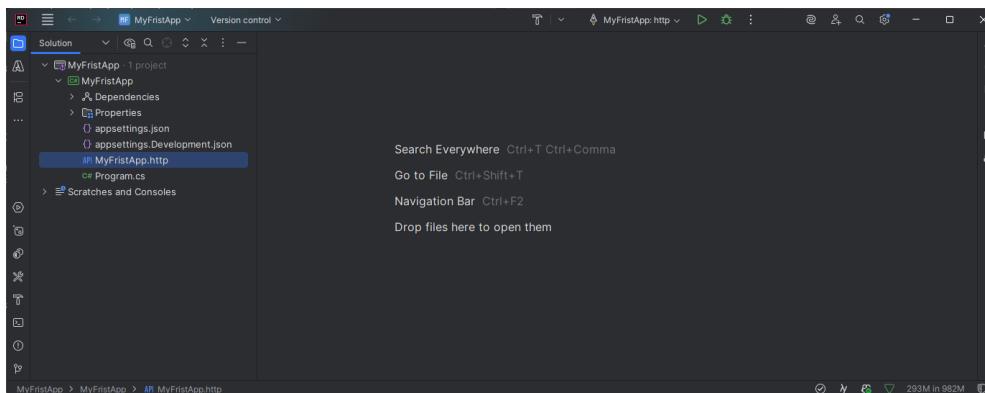


2. Selecciona el tipo de aplicación, en este caso WEB con Témplate WEB API:

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

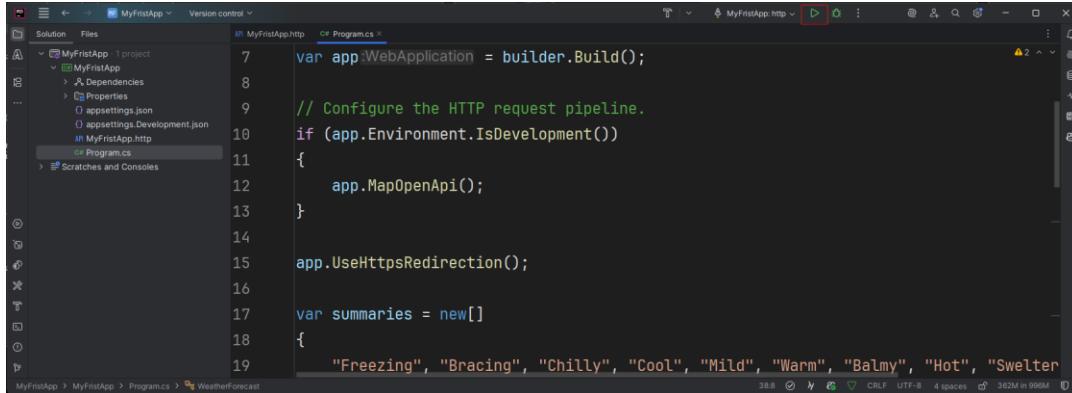


3. Clic en Create.



4. Ejecuta la aplicación desde el IDE:

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025



```

var app = builder.Build();

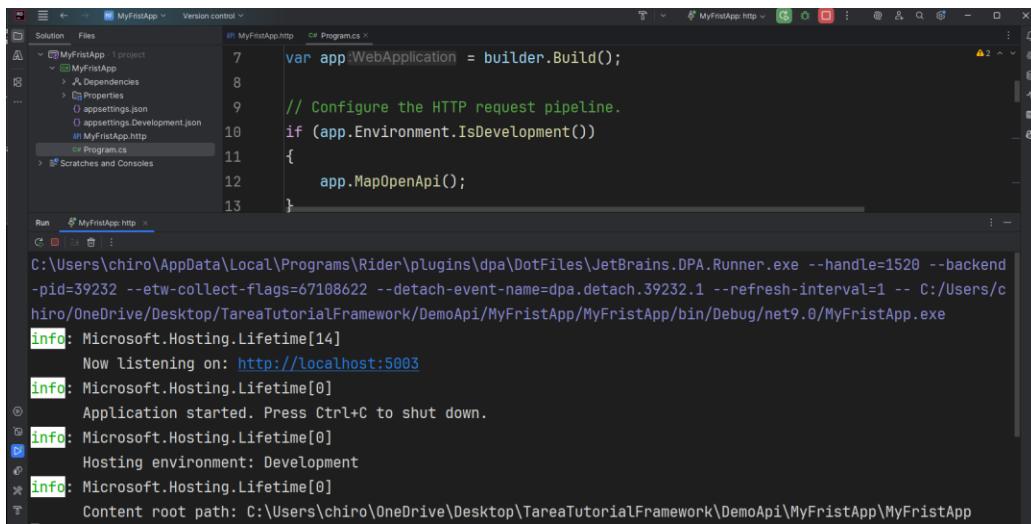
// Configure the HTTP request pipeline.
if (app.Environment.IsDevelopment())
{
    app.MapOpenApi();
}

app.UseHttpsRedirection();

var summaries = new[]
{
    "Freezing", "Bracing", "Chilly", "Cool", "Mild", "Warm", "Balmy", "Hot", "Sweltering", "Scorching"
};

```

5. En el panel run del IDE, podrás encontrar la dirección para visualizar la ejecución (en mi caso <http://localhost:5003>):



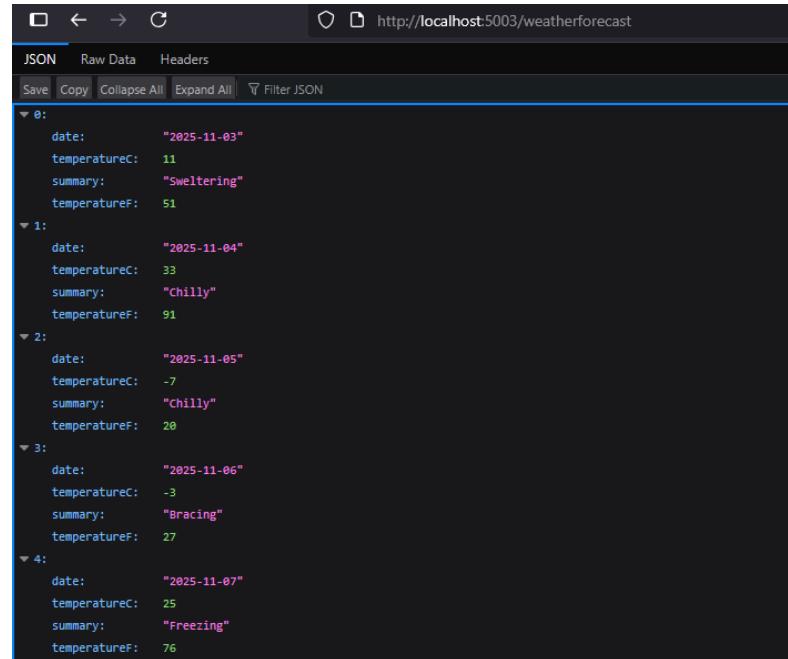
```

C:\Users\chiro\AppData\Local\Programs\Rider\plugins\dpa\DotFiles\JetBrains.DPA.Runner.exe --handle=1520 --backend
-pid=39232 --etw-collect-flags=67108622 --detach-event-name=dpa.detach.39232.1 --refresh-interval=1 -- C:/Users/chiro/OneDrive/Desktop/TareaTutorialFramework/DemoApi/MyFristApp/MyFristApp/bin/Debug/net9.0/MyFristApp.exe
info: Microsoft.Hosting.Lifetime[14]
Now listening on: http://localhost:5003
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: C:/Users/chiro/OneDrive/Desktop/TareaTutorialFramework/DemoApi/MyFristApp/MyFristApp

```

6. Navega el EndPoint REST creado por defecto: <http://localhost:5003/weatherforecast>

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025



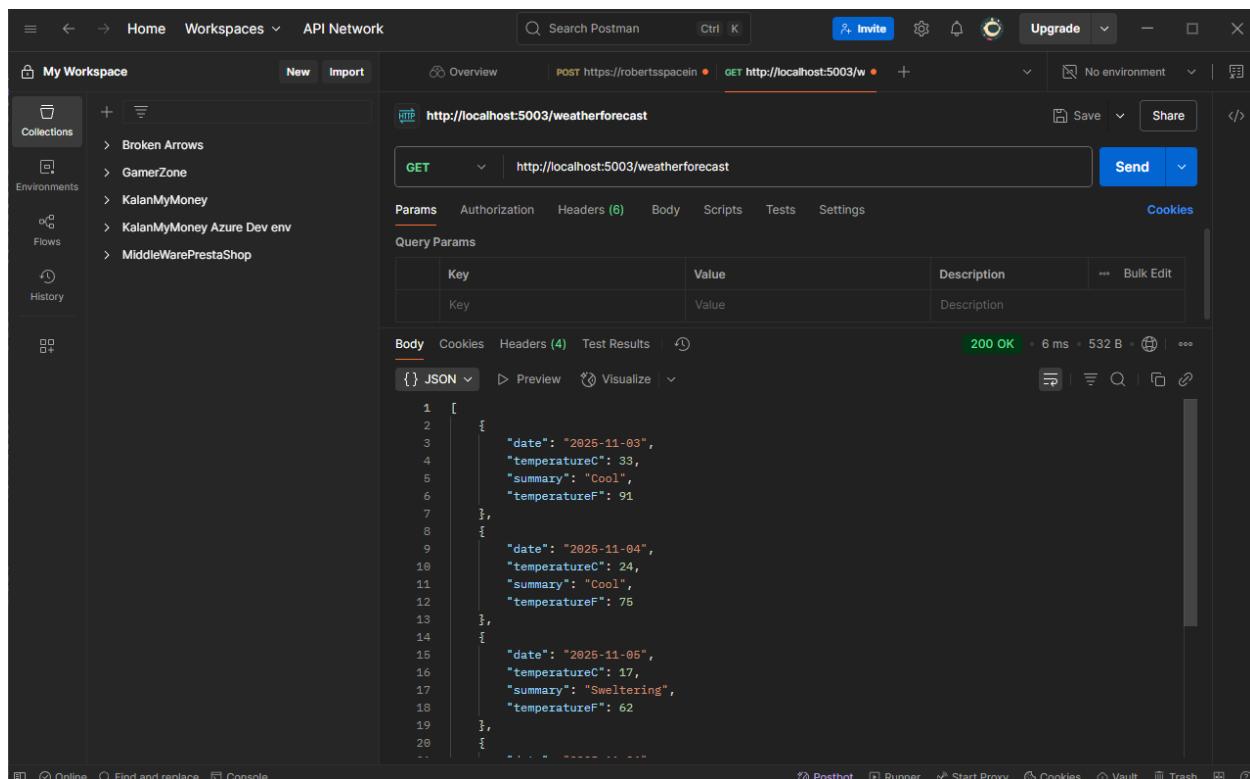
The screenshot shows a JSON object with five elements (0 to 4). Each element contains a date, temperature in Celsius, summary, and temperature in Fahrenheit.

```

{
  "0": {
    "date": "2025-11-03",
    "temperatureC": 11,
    "summary": "Sweltering",
    "temperatureF": 51
  },
  "1": {
    "date": "2025-11-04",
    "temperatureC": 33,
    "summary": "Chilly",
    "temperatureF": 91
  },
  "2": {
    "date": "2025-11-05",
    "temperatureC": -7,
    "summary": "Chilly",
    "temperatureF": 20
  },
  "3": {
    "date": "2025-11-06",
    "temperatureC": -3,
    "summary": "Bracing",
    "temperatureF": 27
  },
  "4": {
    "date": "2025-11-07",
    "temperatureC": 25,
    "summary": "Freezing",
    "temperatureF": 76
  }
}

```

7. Alternativamente puedes usar una herramienta como Postman:



The screenshot shows the Postman interface with a GET request to `http://localhost:5003/weatherforecast`. The response status is 200 OK, and the response body is a JSON array of five objects, each representing a weather forecast for a specific day.

```

[{"date": "2025-11-03", "temperatureC": 11, "summary": "Sweltering", "temperatureF": 51}, {"date": "2025-11-04", "temperatureC": 33, "summary": "Chilly", "temperatureF": 91}, {"date": "2025-11-05", "temperatureC": -7, "summary": "Chilly", "temperatureF": 20}, {"date": "2025-11-06", "temperatureC": -3, "summary": "Bracing", "temperatureF": 27}, {"date": "2025-11-07", "temperatureC": 25, "summary": "Freezing", "temperatureF": 76}]

```

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

5. Explicación del funcionamiento del framework

¿Actúa sobre páginas, ficheros, secciones...?

ASP.NET Core (el framework web de .NET 9) actúa sobre **endpoints HTTP** definidos en **controladores o rutas minimalistas**, no sobre páginas (como en ASP.NET Web Forms).

Cada controlador procesa peticiones HTTP (GET, POST, PUT, DELETE) y devuelve respuestas en formato **JSON**.

¿Cómo se editan/modifican los ficheros?

Los archivos de código fuente se editan directamente desde el IDE o editor de texto. Los más importantes son:

- Program.cs: punto de entrada de la aplicación.
- appsettings.json: configuración (puertos, conexiones, variables, etc.).
- Controllers/*.cs: lógica de los endpoints.

¿Cómo se referencian desde otras aplicaciones?

Las APIs pueden ser consumidas por:

- Aplicaciones web (React, Angular, Blazor).
- Aplicaciones móviles (MAUI, Flutter, Android, iOS).
- Otros servicios (microservicios, colas de mensajes, etc.)

Solo es necesario conocer la **URL base** y los **endpoints** expuestos.

6. Conclusiones

Al finalizar este curso introductorio, el estudiante habrá logrado:

- Comprender **qué es y para qué sirve .NET 9**.
- Instalar correctamente el entorno de desarrollo.

Asignatura	Datos del Equipo	Fecha
Desarrollo de Aplicaciones en Red	Jesús Alberto Castillo Duarte Gonzalo Xochipilli Flores Valdes Vicente Ulises Olivares Plasencia Jorge Alejandro Núñez Delgado Ysmael Paredes Gonzalez Alexia Elizabeth Yáñez Hidalgo	02/11/2025

- Crear y ejecutar su **primera API funcional**.
- Entender cómo se estructuran y configuran los proyectos en .NET.
- Manipular **controladores, rutas y configuraciones** básicas.
- Visualizar resultados mediante **Swagger UI** o herramientas externas.

En resumen, el alumno podrá desarrollar **servicios REST modernos y escalables** con .NET 9, listos para integrarse en arquitecturas empresariales o en la nube (Azure, AWS, GCP).