



UNIVERSIDAD DE CUENCA

MANUAL DE DESPLIEGUE PARA CRUD USANDO BACK-END Y FRONT-END

Marisol Peñafiel
Michael Peñaloza
Oscar Merino
Andres Meneses

BACKEND

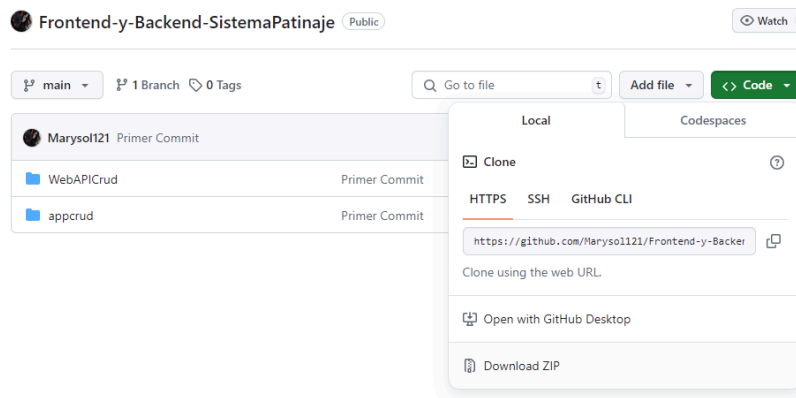
1. Inicio

Como paso esencial es tener las instalaciones necesarias que son ASP.NET CORE, Microsoft Visual Studio y Microsoft SQL Server Management Studio, dichas instalaciones ya se hicieron como parte del trabajo de los Frameworks.

(En caso de no tener las instalaciones necesarias, se puede revisar el documento adjunto "Instalaciones_Necesarias_ASPNetCore.pdf").

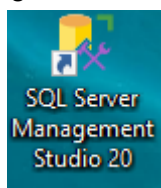
2. Descarga de proyecto:

- En el siguiente link de Github:
<https://github.com/Marysol121/FB-SistemaPatinaje.git>
nos vamos a descargar y descomprimir el archivo del proyecto.

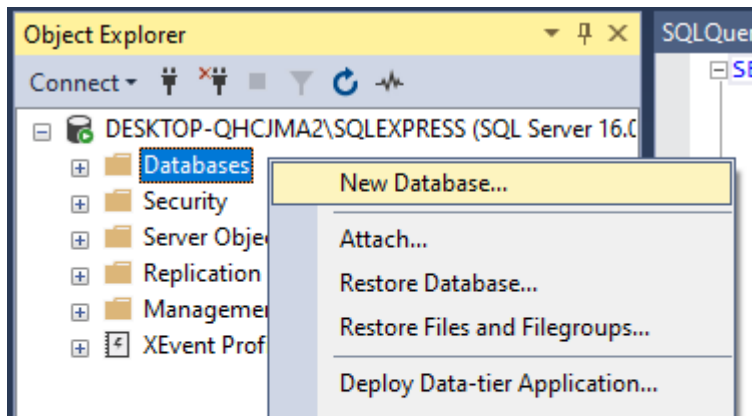


3. Creación de base de datos:

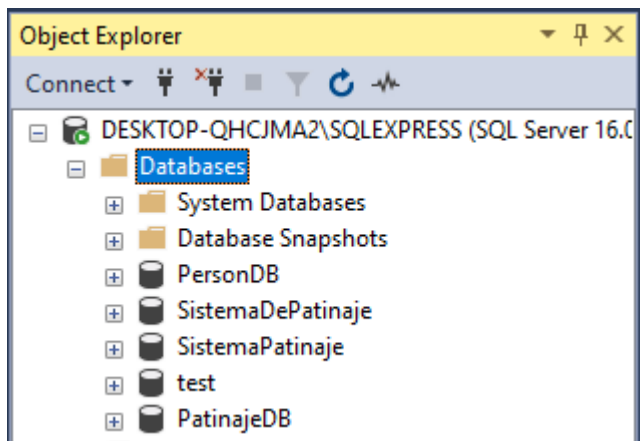
- Abrimos Microsoft SQL Server Management Studio



- Procedemos a crear una base de datos dando click derecho sobre “Databases” y seleccionando “New Database”

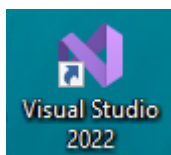


- y le damos el nombre de “PatinajeDB” a la nueva base de datos. pulsamo OK y verificamos que se haya creado:

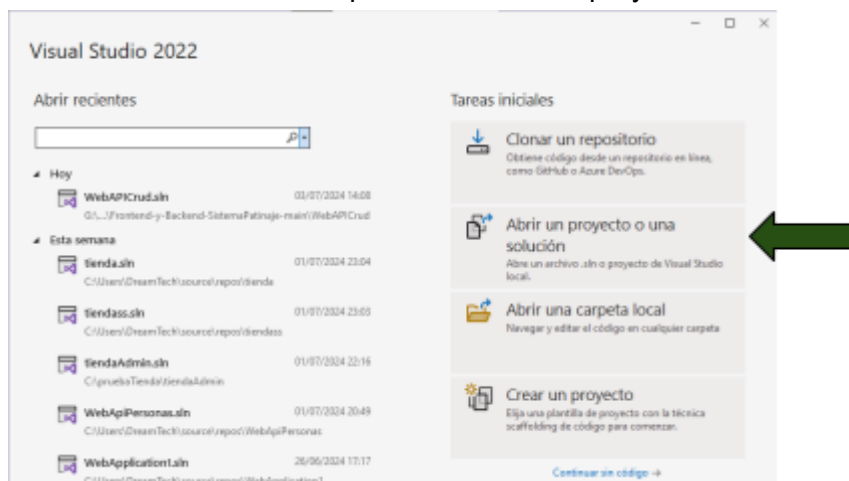


4. Proyecto:

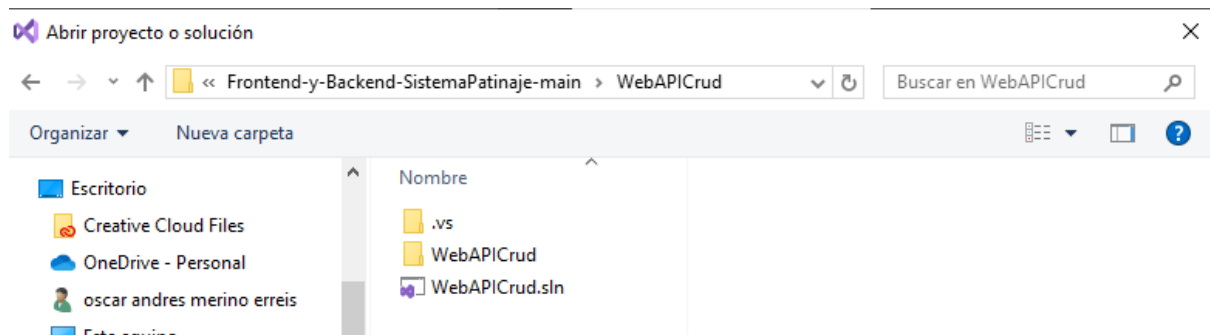
- Se abre Microsoft visual Studio



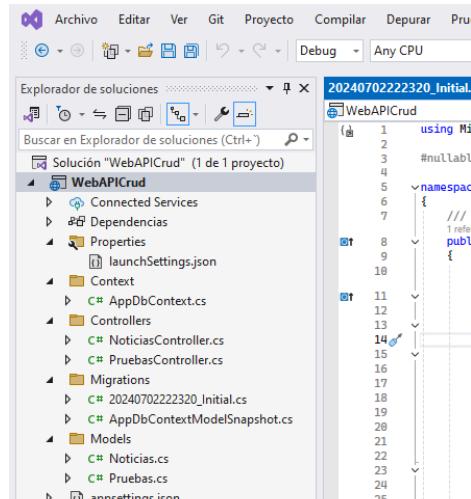
- seleccionamos la opción de “Abrir un proyecto o una solución”



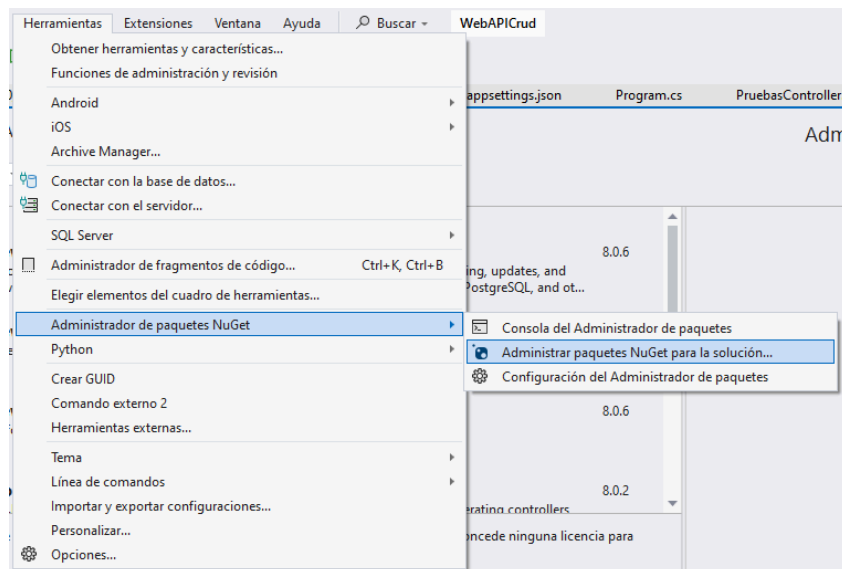
y navegamos hasta la carpeta del proyecto que descargamos abrimos “WebAPICrud”, seleccionamos el archivo WebAPICrud.sln y le damos OK/Abrir:



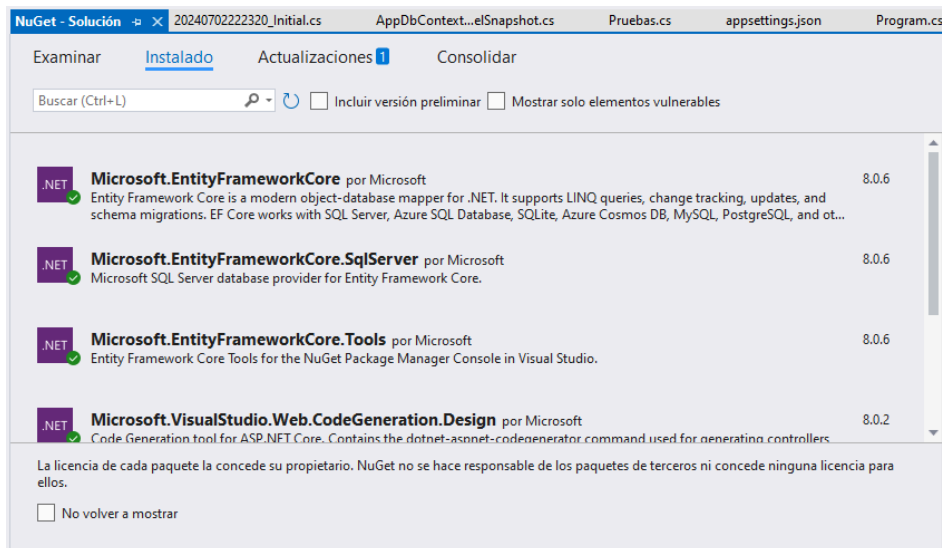
Nos debería aparecer todas las carpetas y archivos del proyecto:



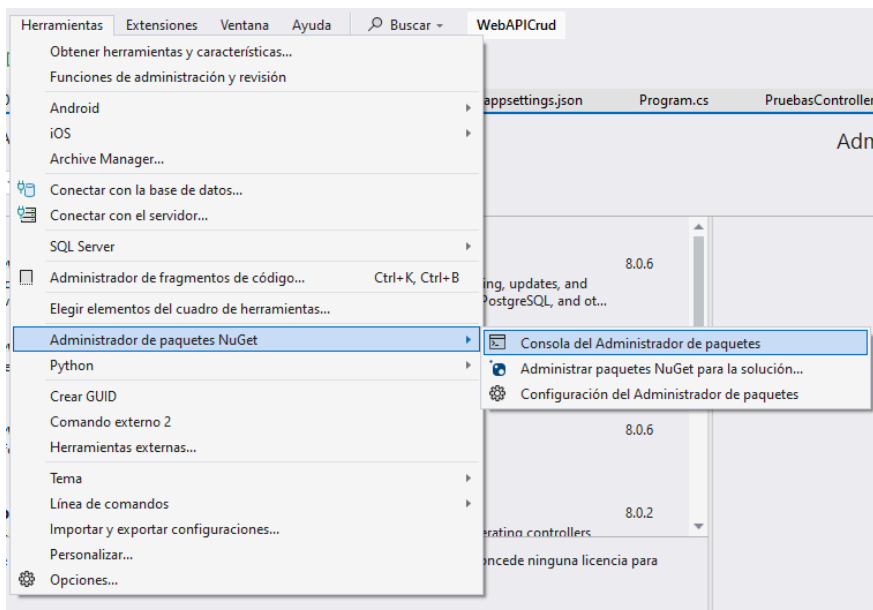
- En la parte superior en herramientas/tools vamos a Administrador de paquetes NuGet > Administrar paquetes nuget para la solución



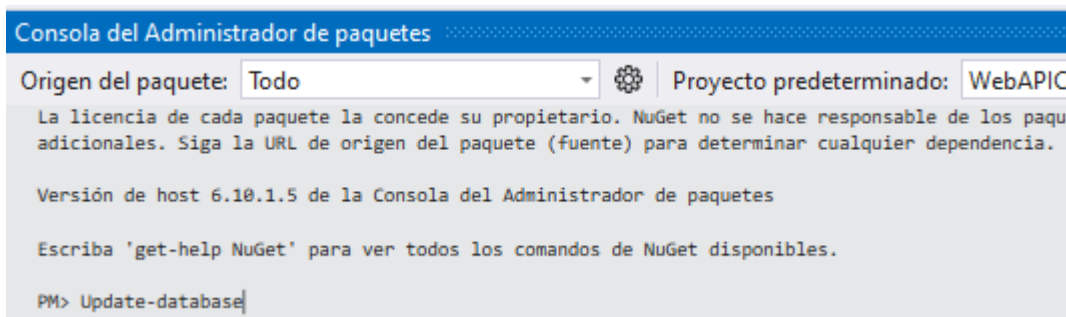
Chequeamos en la pestaña instalado que estén instalados los siguientes paquetes, y si no en la pestaña Examinar los podemos instalar:



- volvemos nuevamente a la pestaña de herramientas/tools pero esta vez seleccionamos Consola de administrador de paquetes:

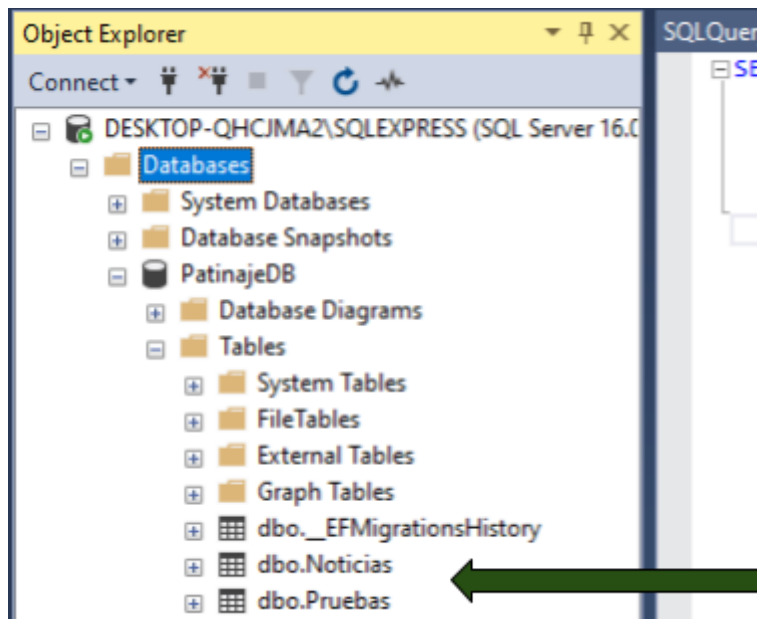


en la consola que se nos ha abierto ingresamos **“Update-database”**

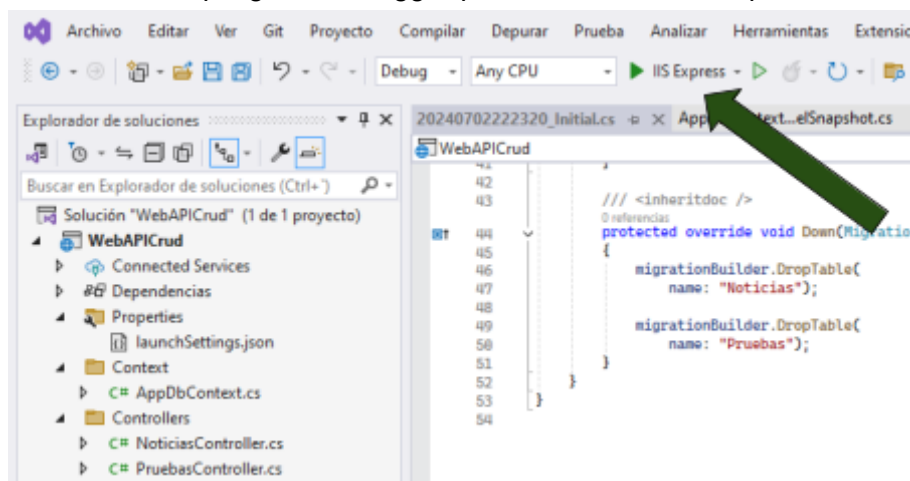


y esperamos que se ejecute.

- Luego en la aplicación de Microsoft SQL Server Management Studio podemos verificar que se han creado las tablas Noticias y Pruebas:

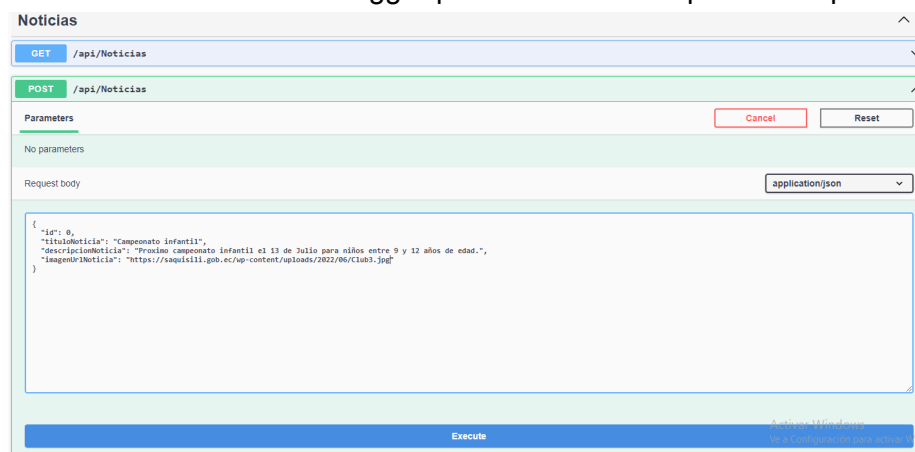


- Finalmente ejecutamos la aplicación seleccionando "IIS Express" en caso de que no nos despliegue el Swagger podemos cambiar la opción "IIS Express" por "https":



5. Swagger UI

Una vez se nos abre el Swagger podemos hacer un post como prueba:



Ejecutamos:

Responses

Curl

```
curl -X 'POST' \
  https://localhost:44329/api/Noticias \
  -H 'accept: text/plain' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 0,
    "tituloNoticia": "Campeonato Infantil",
    "descripcionNoticia": "Proximo campeonato infantil el 13 de Julio para niños entre 9 y 12 años de edad.",
    "imagenNoticia": "https://saquilil.gob.ec/wp-content/uploads/2022/06/Club3.jpg"
  }'
```

Request URL

https://localhost:44329/api/Noticias

Server response

Code

Details

201

Indocumented

Response body

```
{
  "id": 1,
  "tituloNoticia": "Campeonato Infantil",
  "descripcionNoticia": "Proximo campeonato infantil el 13 de Julio para niños entre 9 y 12 años de edad.",
  "imagenNoticia": "https://saquilil.gob.ec/wp-content/uploads/2022/06/Club3.jpg"
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Wed, 03 Jul 2024 20:16:07 GMT
location: https://localhost:44329/api/Noticias/1
server: Microsoft-IIS/10.0
x-powered-by: ASP.NET
```

Responses

Code

Description

Links

200

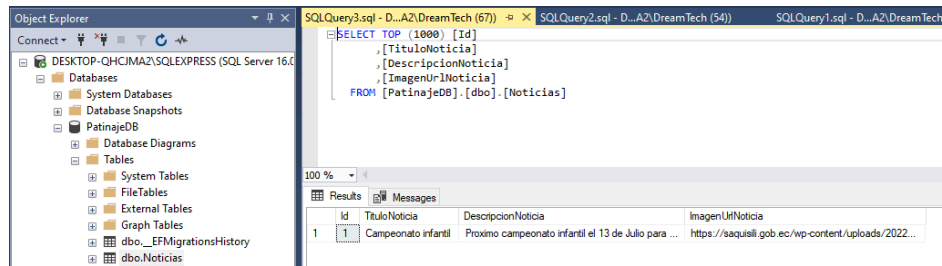
Success

Activar Windows

Ve a Configuración para activar Windows

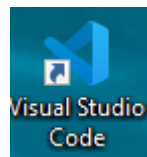
No links

y podemos verificar en la Base de datos que se ha ingresado una noticia correctamente:

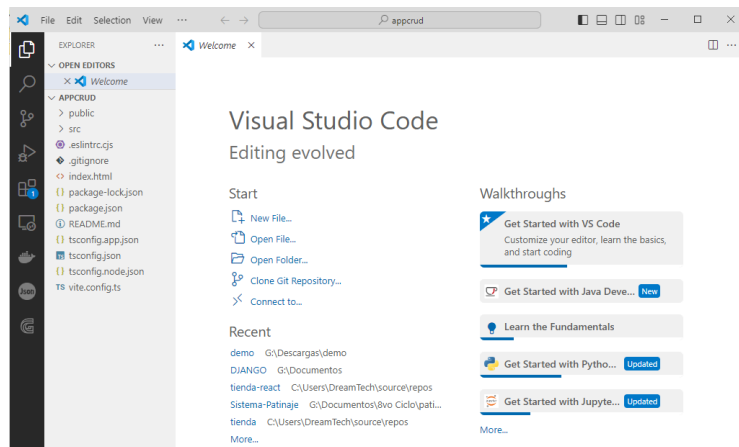


FRONTEND

1. Para trabajar con react en nuestro frontend primero necesitamos tener instalado Node.js, si no lo tiene se puede descargar del siguiente enlace: <https://nodejs.org/en> y el IDE Visual Studio Code:



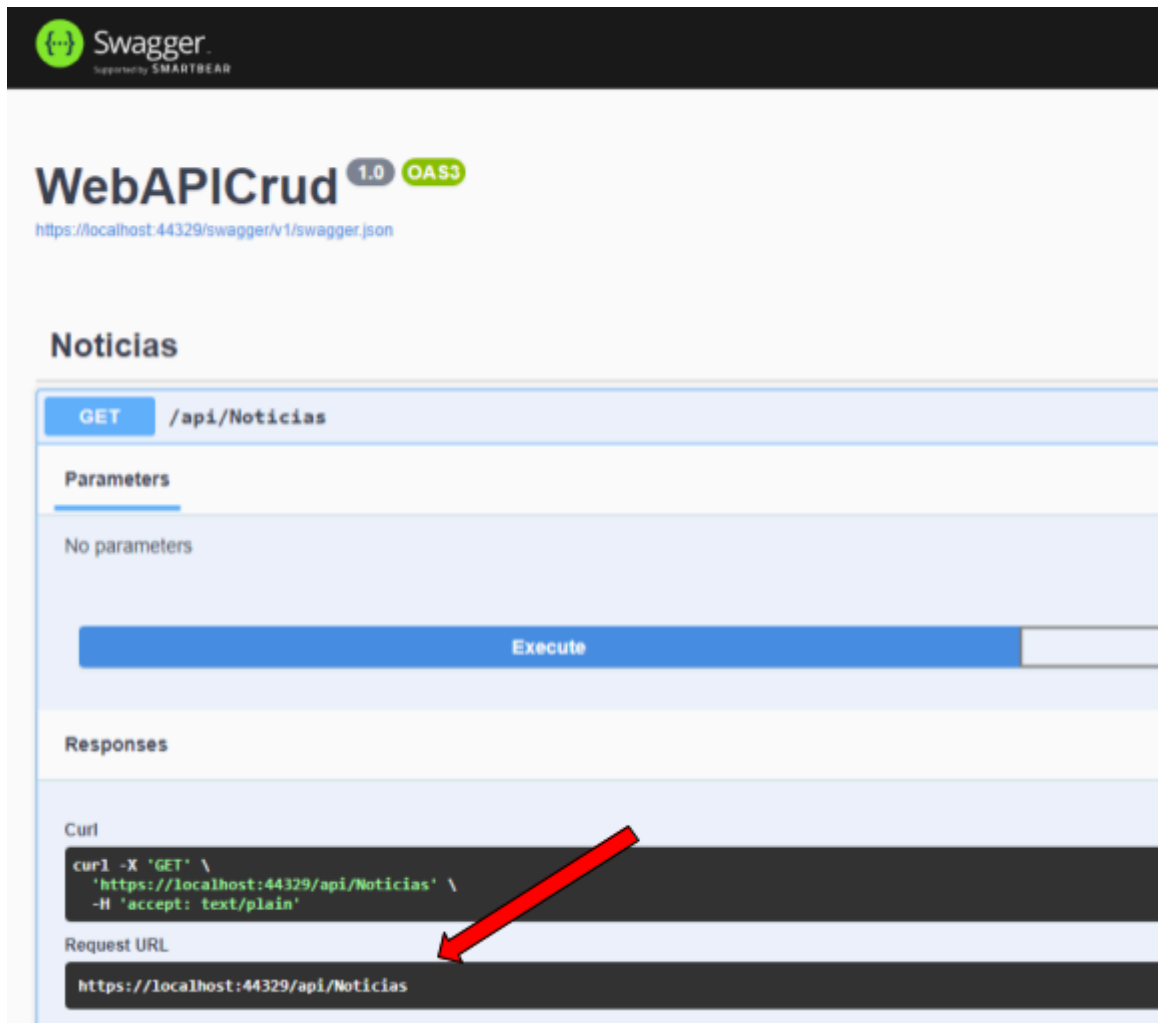
2. Abrimos en Visual Studio Code la carpeta "appcrud" que se encuentra en la carpeta del proyecto que descargamos: quedándonos de la siguiente manera:



3. abrimos una terminal y corremos el comando "npm install" y esperamos se instale:

```
PS G:\Descargas\Frontend-y-Backend-SistemaPatinaje-main\Frontend-y-Backend-SistemaPatinaje-main\appcrud> npm install
```

4. ¡MUY IMPORTANTE! en el swagger cuando se hace una petición GET por ejemplo nos da una dirección:



5. copiamos la dirección y la reemplazamos por la otra dirección que está en el archivo src/settings/appsettings.ts en visual studio code:

```
evaNoticia.tsx  EditarNoticia.tsx  react.svg  TS appsettings:
src > settings > TS appsettings.ts > appsettings
1  export const appsettings = {
2    apiUrl: "https://localhost:44329/api/Noticias"
3  }
```

borramos los últimos caracteres de la dirección hasta /api/

```
evaNoticia.tsx  EditarNoticia.tsx  react.svg  TS appsetting
src > settings > TS appsettings.ts > appsettings > apiUrl
1  export const appsettings = {
2    apiUrl: "https://localhost:44329/api/"
3  }
```

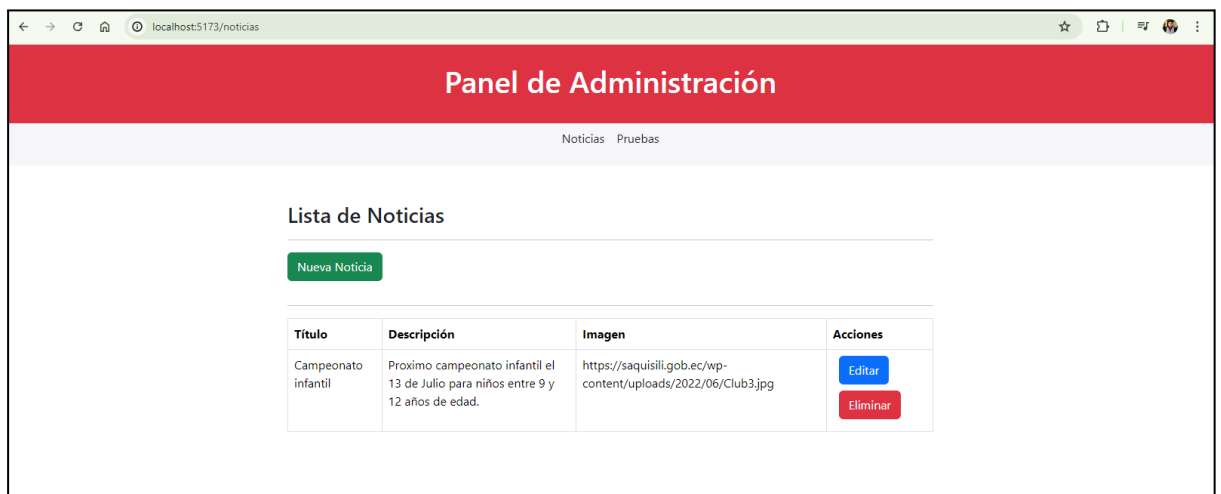

7. Finalmente en la terminal de visual studio code corremos el siguiente comando “npm run dev” esperamos y se nos dará una dirección http para ingresar a nuestro servidor mediante un navegador:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

VITE v5.3.2  ready in 542 ms
→ Local:   http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
```

y lo que observaremos en nuestro navegador será algo como la siguiente pantalla:



Una vez desplegado el crud en el front end se puede ingresar nuevas noticias editarlas las existentes o eliminar noticias.