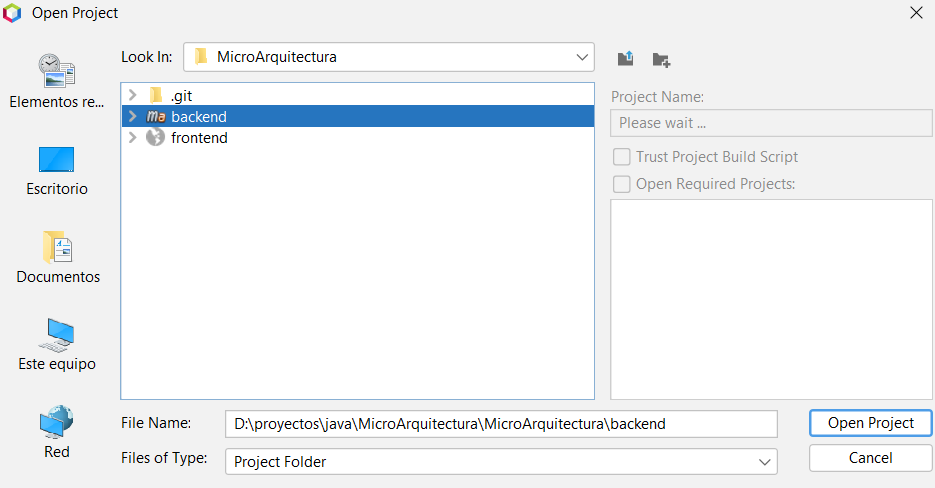
**Proyecto Micro Arquitectura**

**Backend**: Java Sprint Boot – Micro servicios – Rest Full

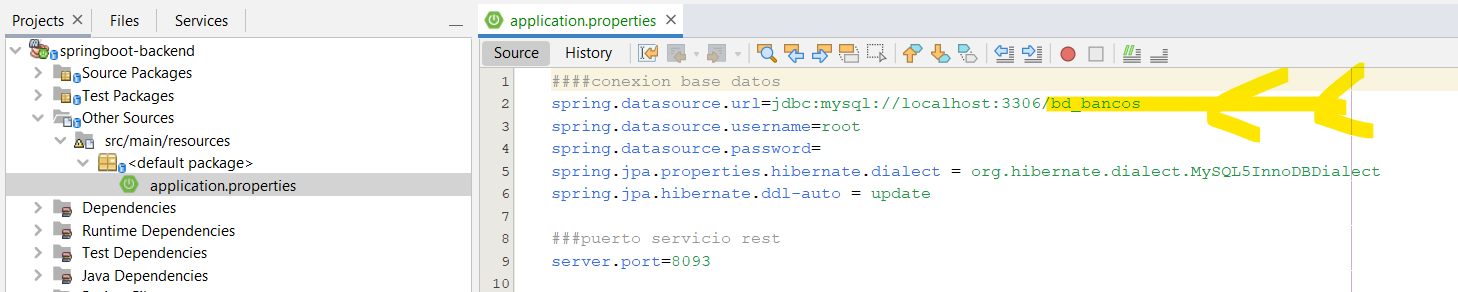
**Frontend**: Javascript React

**BACKEND – MICRO SERVICIOS**

* En el editor NETBEANS, abrir el Proyecto **Backend**.

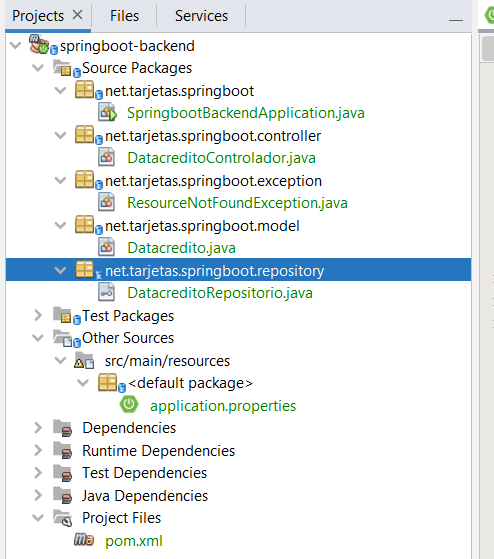


* Abrir el fichero “application.properties“ y cambiar la conexión a base de datos si es el caso.



El proyecto está organizado según el estándar MVC, donde la lógica del negocio está separada en componentes.

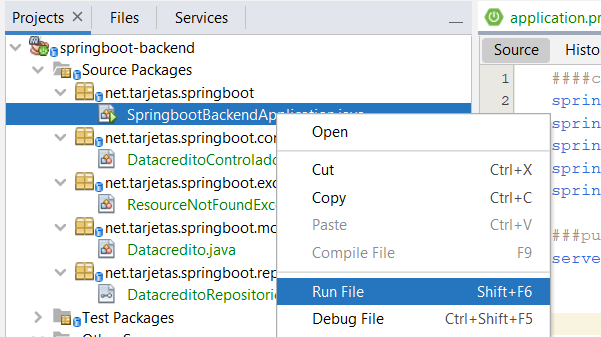
Como es un micro servicio solo tiene un controlador.



* Antes de ejecutar el micro servicio, validar que la instancia de Base datos este en ejecución.

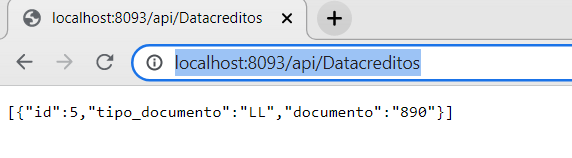


* Para ejecutar el proyecto, seleccionamos el fichero “SpringbootBackendApplication.java” y le damos click derecho, correr fichero.



Para validar que está corriendo el micro servicio, vamos al navegador o a “insomnia” y probamos la dirección donde se listan los datos:

<http://localhost:8093/api/Datacreditos>



En estos momentos el micro servicio es **funcional**.

**FRONTEND – MICRO APLICACIONES – MICRO FRONTEND**

Teniendo el micro servicio ejecutándose.

Vamos a proceder a ejecutar el **frontend**.

Descargamos e instalamos el ultimo **NodeJs** disponible

<https://nodejs.org/dist/v18.10.0/node-v18.10.0-x64.msi>

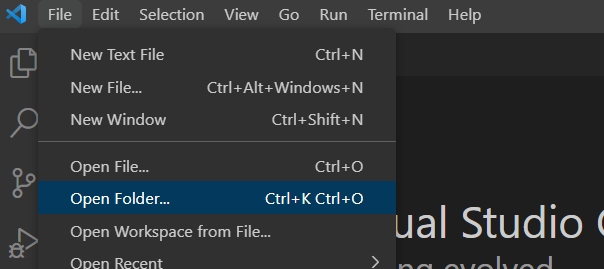
o del listado

<https://nodejs.org/en/download/current/>

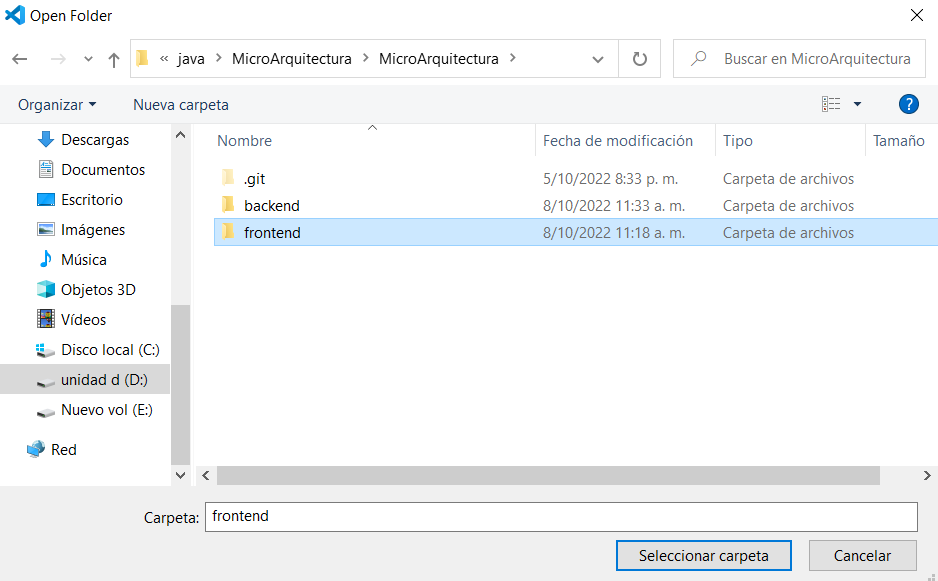
Descargamos e instalamos el **Visual Studio Code**

<https://code.visualstudio.com/download>

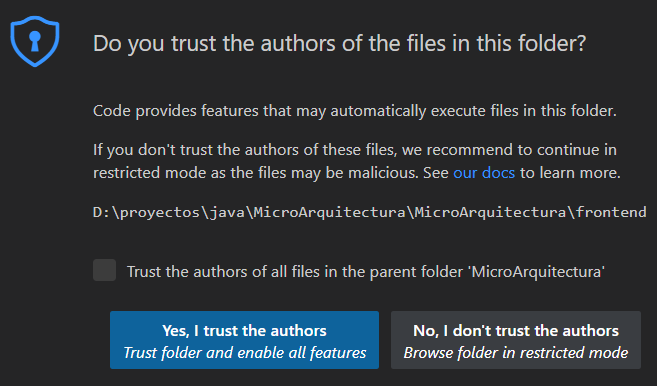
* Vamos al menú archivo, luego menú Abrir directorio.



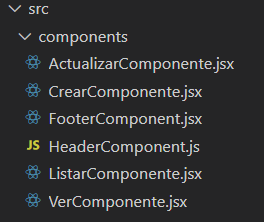
* Abrimos el directorio Frontend



* Si aparece un mensaje de advertencia, le damos que “**si confiamos en los autores**“



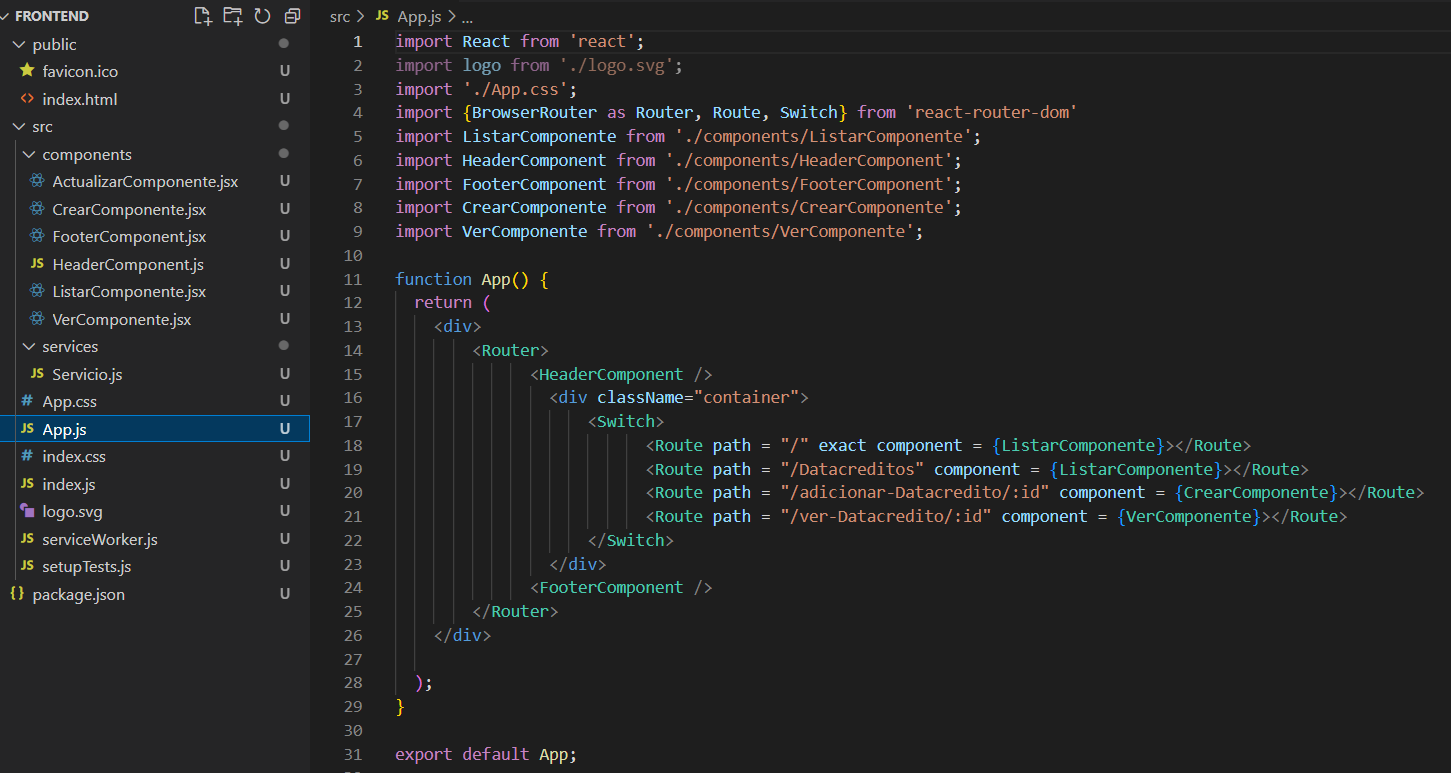
* El micro frontend está organizado por componentes, que son las sub páginas y controles, que conforman la micro aplicación.



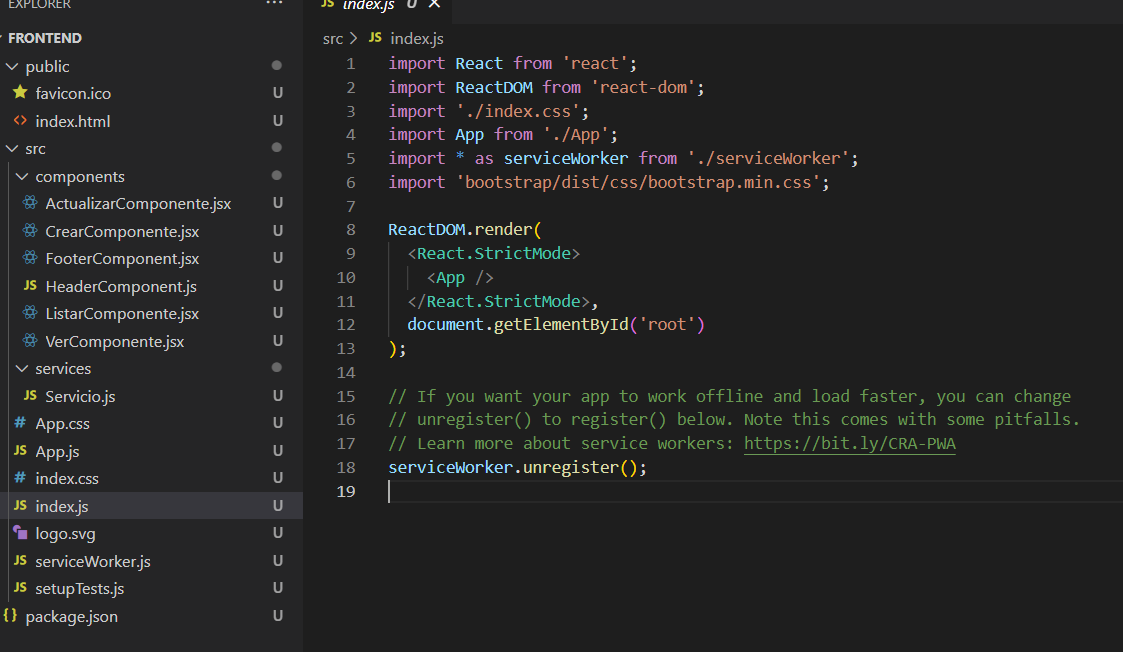
* Los servicios, que nos sirven para separar la lógica del negocio, y como interactuamos con los micro servicios



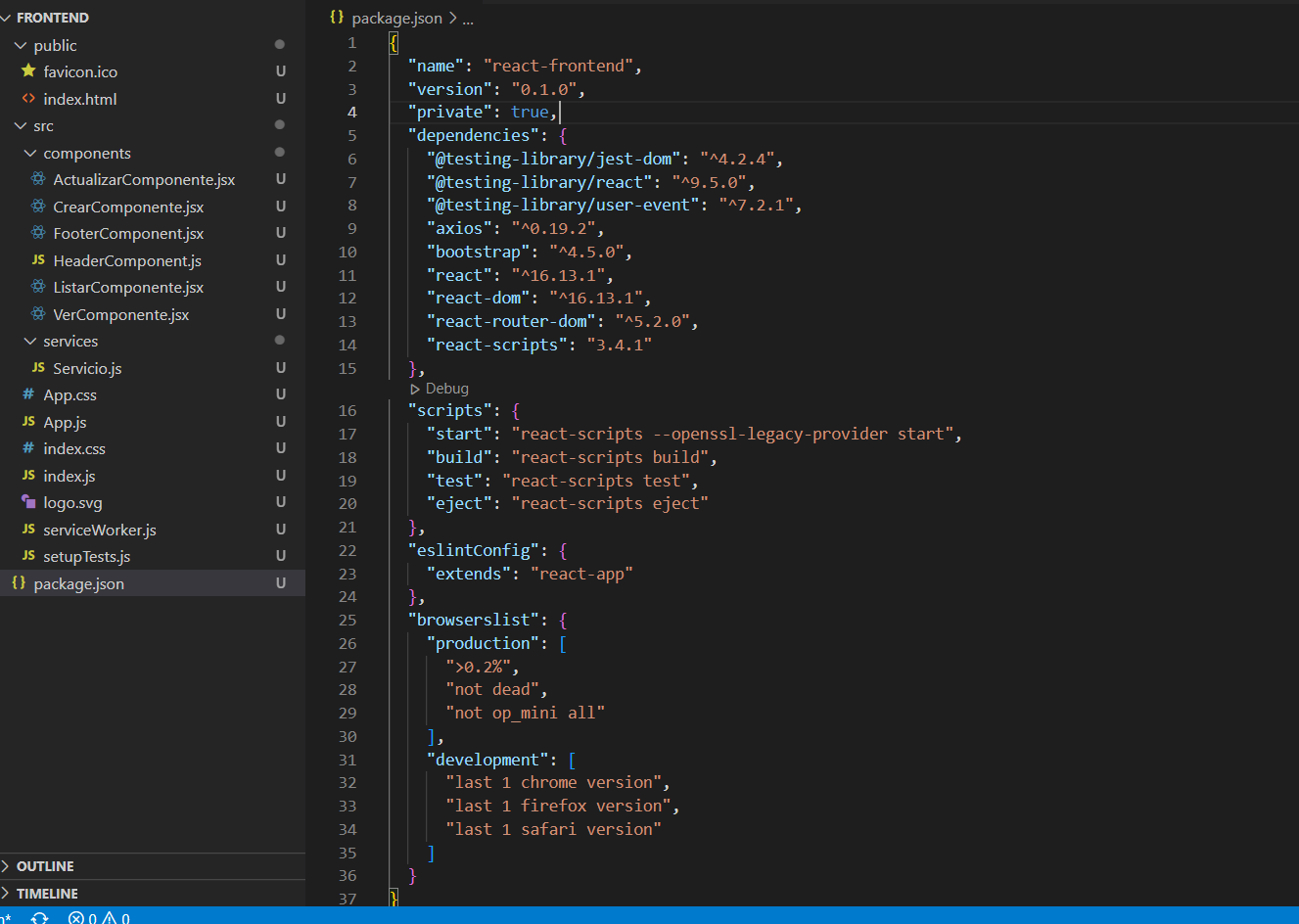
* El fichero app.js, donde se define el cómo se comporta el contenedor y el enrutamiento de las sub páginas, que conforman la micro aplicación, por ejemplo, como se navega con las sub paginas listar, crear, actualizar, borrar, etc. y aquí se define cual seria el index.html que en este caso sería el carácter raíz de directorio “/”.



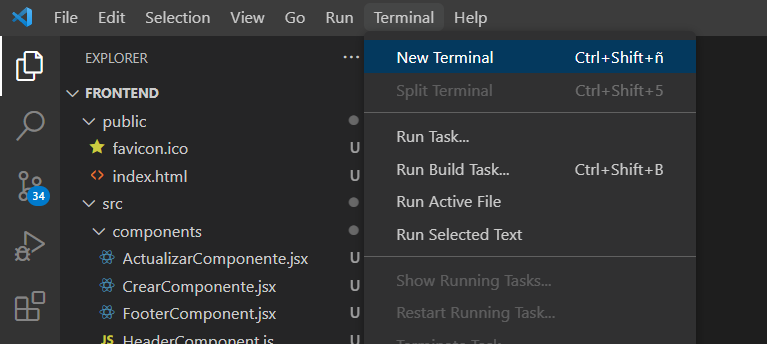
* El fichero index.js, que es la aplicación principal que contiene y llama a todas las sub páginas, aquí se define el tag html <App /> donde se renderiza todos los componentes.



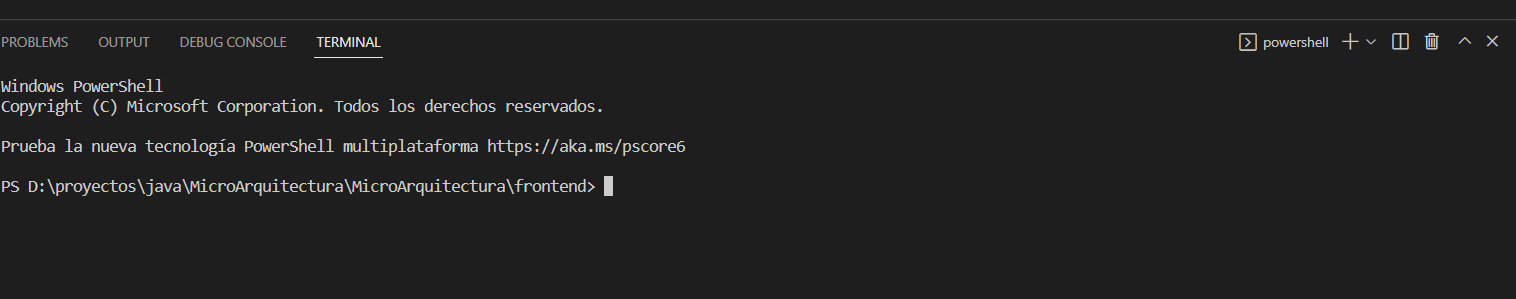
* El fichero package.json, donde se definen todas las dependencias del proyecto de micro frontend en React. Y como se comporta el proyecto cuando lo corremos(start), compilamos (build), etc.



* Ya conociendo la estructura del micro frontend, procedemos a correrlo.
* Vamos al menú terminal, luego menú nuevo terminal.



* Vemos que se abre un terminal, en la parte inferior derecha. Similar a la consola DOS o POWERSHELL de Windows.



* En la consola digitamos

**npm install**

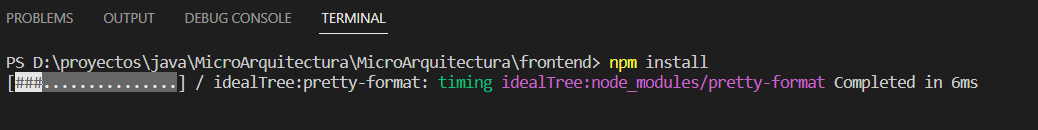
Si hay errores, en el anterior comando, ejecutamos los siguientes comandos, uno a la vez:

**npm config set fund false –global**

**npm audit fix –force**

**npm install**

si no hay errores, se visualiza una descarga de componentes similar a esto.



* Luego de terminar el proceso, debemos iniciar el proyecto con el siguiente comando.

**npm start**

* En el navegador digitamos la dirección URL, siguiente:

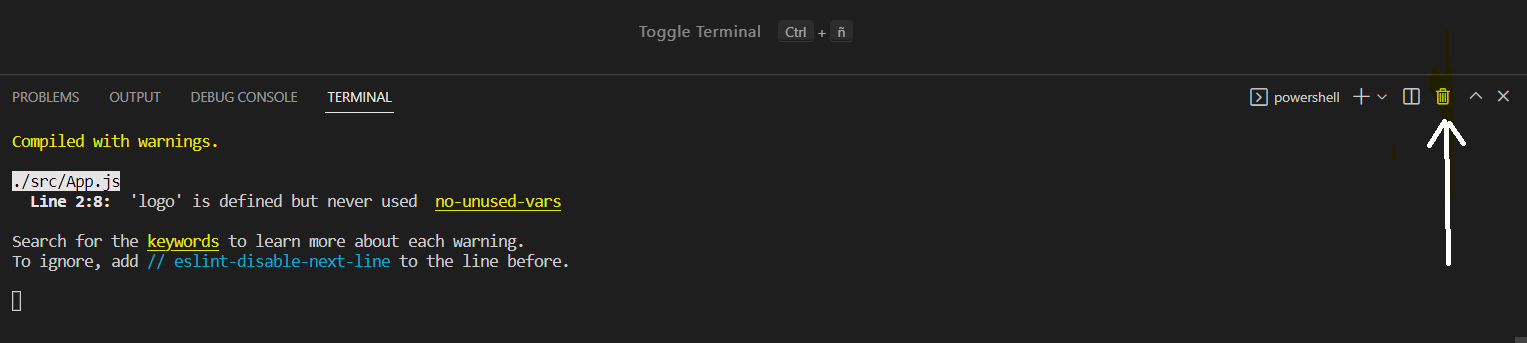
<http://localhost:3000>

Si todo está correcto, se debe visualizar el micro frontend en el navegador



En estos momentos la aplicación es **funcional**.

Para detener la ejecución del micro frontend, solo basta presionar el icono de papelera en la terminal.



Y para volverla a ejecutar

**npm start**

Si se desean realizar cambios en el frontend, no hay necesidad de reiniciar la aplicación, los cambios se ven reflejados al instante, siempre y cuando se guarden los cambios en VsCode.