

- [Pasos para conectar React con la API en Laravel](#)
- [Resumen](#)

Para conectar tu aplicación **React** con una API en Laravel, puedes usar [fetch](#) o [axios](#).

Pasos para conectar React con la API en Laravel

1. Configurar CORS en Laravel

En [app/Http/Middleware/HandleCors.php](#) asegúrate de permitir las solicitudes desde el frontend:

```
public function handle($request, Closure $next)
{
    return $next($request)
        ->header('Access-Control-Allow-Origin', '*')
        ->header('Access-Control-Allow-Methods', 'GET, POST, PUT, DELETE,
OPTIONS')
        ->header('Access-Control-Allow-Headers', 'Content-Type,
Authorization');
}
```

Si usas Laravel 11+, puedes configurar esto en [config/cors.php](#).

2. Instalar Axios en React

```
npm install axios
```

3. Hacer una petición desde React a Laravel

```
import { useEffect, useState } from "react";
import axios from "axios";

function App() {
  const [data, setData] = useState([]);

  useEffect(() => {
    axios.get("http://127.0.0.1:8000/api/notes")
      .then(response => setData(response.data))
      .catch(error => console.error(error));
  }, []);

  return (
    <div>
      {data.map(note => (
        <div>
          {note.title}
          {note.content}
        </div>
      ))}
    </div>
  );
}

export default App;
```

```

        .then(response => {
            setData(response.data);
        })
        .catch(error => {
            console.error("Error al obtener datos:", error);
        });
    }, []);
}

return (
    <div>
        <h1>Notas</h1>
        <ul>
            {data.map((note) => (
                <li key={note.id}>{note.title}</li>
            )))
        </ul>
    </div>
);
}

export default App;

```

4. Envío de datos a Laravel desde React

```

const enviarNota = async () => {
    try {
        const response = await axios.post("http://127.0.0.1:8000/api/notes", {
            title: "Nueva Nota",
            content: "Contenido de la nota"
        });
        console.log("Nota creada:", response.data);
    } catch (error) {
        console.error("Error al enviar nota:", error);
    }
};

```

5. Asegurar autenticación con Sanctum

Si usas **Sanctum** en Laravel para autenticación, debes configurar **withCredentials: true** en **axios**:

```
axios.defaults.withCredentials = true;
```

Resumen

- Laravel debe permitir CORS.
- React puede hacer peticiones con `axios` o `fetch`.
- Si usas autenticación, configura `Sanctum` correctamente.

🔗 **Prueba en Postman** primero para verificar que la API funciona antes de conectarla con React. 🚀