

CIFP Carlos III

# Proyecto fullstack node + React

Desarrollo web en entorno cliente

Alejandro Martínez Bravo  
19-2-2026

# Memoria Técnica: Aplicación Meteorológica AEMET

## 1. Descripción del Proyecto

Se ha desarrollado una aplicación web Full Stack que consume datos en tiempo real de la Agencia Estatal de Meteorología (AEMET). La plataforma permite a los usuarios consultar predicciones meteorológicas mediante tres vías de interacción:

- **Búsqueda por provincia:** Filtrado jerárquico de datos.
- **Búsqueda por municipio:** Selección específica de localidades.
- **Búsqueda directa por ID:** Introducción manual del código de municipio para usuarios avanzados que conozcan el código.

El sistema se divide en un Backend que actúa como proxy para gestionar la seguridad y las peticiones a la API oficial, y un Frontend dinámico que presenta la información de forma clara al usuario.

## 2. Decisiones Técnicas

- **Arquitectura de Backend (Node.js):** Permite tener una estructura modular que facilita el desarrollo y posible mantenimiento. Se utilizaron:
  - **dotenv:** Para gestionar la API KEY de forma segura, evitando exponerla en el código fuente.
  - **cors:** Para permitir la comunicación segura entre el servidor de frontend y el de backend.
  - **Proxy de API:** El backend no solo sirve datos, sino que gestiona la lógica de las peticiones a AEMET, simplificando la respuesta para el frontend.
- **Arquitectura de Frontend (React):**
  - **Custom Hooks:** Se implementó useFetch para centralizar la lógica de las peticiones asíncronas.
  - **Optimización con AbortController:** Se incluyó un sistema de cancelación de peticiones para evitar fugas de memoria y asegurar que, si el usuario navega rápido entre municipios, solo se procese la última petición solicitada.
  - **Componentización:** La interfaz se dividió en componentes reutilizables (button, input, dropdown) para seguir la filosofía de React.

- **Control de errores:** Implementados avisos en consola por si algo falla, saber el qué.
- **Datos:** Funciona sin datos locales (excepto un array de provincias, ya que la api no lo proporciona como sí lo hace con los municipios). Siempre recurre a la información reciente que proporciona la AEMET.
- **Gestión de utilidades y organización basada en componentes:** Se separa la funcionalidad de la aplicación de la UI (ej: pronosticoTabla). Se aplica el concepto de composición, por el que se construyen elementos complejos a partir de piezas simples (carpeta con componentes). Funciones puras (como trimId.js) separadas de la lógica de React. Lógica desacoplada que permite la reutilización (useFetch reutilizable para otras posibles secciones u otras llamadas).

### 3. Conclusiones

La realización de este proyecto ha supuesto un gran reto y ha ampliado mis conocimientos sobre tecnologías de desarrollo web:

- **Diseño de APIs:** El mayor reto fue estructurar las rutas del backend y asegurar que el flujo de datos fuera coherente y correcto desde el servidor de AEMET hasta la pantalla del usuario.
- **Aprendizaje de React:** Aunque ha sido un proceso de aprendizaje sobre la marcha, he comprendido el potencial de los estados y efectos para crear interfaces reactivas.
- **Herramientas profesionales y potentes:** El uso de Postman ha sido clave para testear las rutas del backend antes de integrarlas con el frontend, ahorrando tiempo en la depuración.
- **Visión de futuro:** Tras experimentar con Node y React, he descubierto la potencia del desarrollo Full Stack, especialmente la robustez que aporta Node en el lado del servidor (gestiona varias tareas de manera eficiente). Prefiero estas tecnologías más intuitivas que otras vistas en este grado.