

## Diagrama de flujo de navegación y datos

Este diagrama muestra la interacción entre los componentes principales y las peticiones a la API.

Permite visualizar cómo el usuario navega desde la lista de programas hasta las búsquedas y pruebas de funciones.

[Ver diagrama completo de navegación y flujo de datos](#)

## ProgramasAuxiliares.tsx

Este componente lista los **Programas Auxiliares** disponibles y permite navegar a su definición.

Carga inicialmente todos los programas y muestra subprogramas al seleccionar uno.

Cód.	Programa	Acciones
1	Perfil de Barrada	<a href="#">Ir a definición</a>
2	Sugerencias de Colores	<a href="#">Ir a definición</a>
3	Cálculo de PCM y C. Eq.	<a href="#">Ir a definición</a>
4	Peso Complementario	<a href="#">Ir a definición</a>
5	Horas Hombre, MOD y TPM	<a href="#">Ir a definición</a>
6	Lote Economico	<a href="#">Ir a definición</a>
7	Kg por Horas Hombre	<a href="#">Ir a definición</a>
8	Cálculo de Piezas por Hora	<a href="#">Ir a definición</a> <a href="#">Ver todos</a>
9	Consumo Unitario de Materia Prima	<a href="#">Ir a definición</a>
10	Cálculo de Piezas por Hora para Grupos	<a href="#">Ir a definición</a> <a href="#">Ver todos</a>
12	Orden de Compra de Laboratorio Inicial	<a href="#">Ir a definición</a> <a href="#">Ver todos</a>
20	Programa de PBC para PF con chapa	<a href="#">Ir a definición</a> <a href="#">Ver todos</a>
21	Validación de Colada	<a href="#">Ir a definición</a> <a href="#">Ver todos</a>

Figura 1 – Pantalla principal de Programas Auxiliares con listado y acciones ir a definición y ver todos.

### Endpoints

- **GET /programas-auxiliares**

Recupera el listado completo de programas y su estructura padre-hijo.

### Fallas detectadas

- Ninguna lógica crítica detectada.

## ProgramaAuxiliar.tsx

Muestra el **detalle de un Programa Auxiliar**, su definición SQL y permite editar, imprimir o ver su historial.

Integra los subcomponentes de prueba y búsqueda de IDs cuando es la versión vigente.

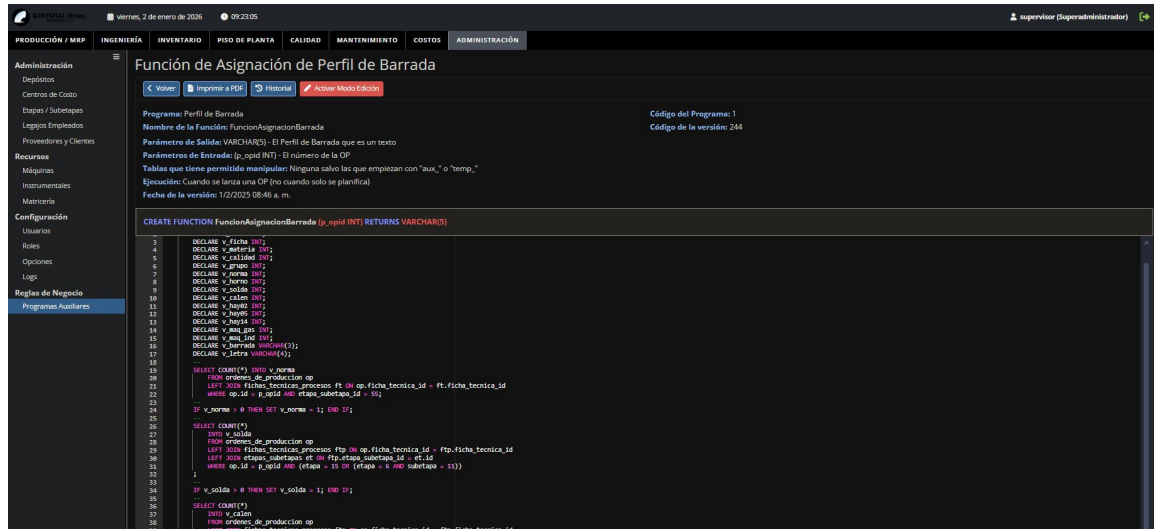


Figura 2 - Vista detallada de un programa auxiliar, con acciones de editar, ver historial, observaciones, probar programa e impresión.

### Endpoints

- **GET** /programa-auxiliar

Obtiene datos del programa según id en query params.

- **PATCH** /programa-auxiliar

Actualiza la definición y las observaciones enviadas en el body.

### Fallas detectadas

- El botón “Volver a esta Versión” tiene la clase d-none y nunca se muestra.
- La función de reactivar versión está comentada, lo que puede confundir si se planeaba usar.

## ProgramaAuxiliarProbar.tsx

Permite **probar** una función SQL del programa auxiliar enviando sus parámetros.

Construye dinámicamente los inputs según la firma (nombre tipo, ...).

### Endpoints

- **GET** /programa-auxiliar/prueba

Envía los pares nombre=valor como query params y devuelve el resultado.

### Fallas detectadas

- Si la función retorna 0, se considera error y muestra mensaje, aunque 0 podría ser un resultado válido.
- La deducción de tipo (int vs otros) solo mira si `tipo.toLowerCase() === 'int'`, no contempla otras tipificaciones.

## ProgramaAuxiliarBusca.tsx

Integra tres vistas de búsqueda de **IDs auxiliares**: ficha técnica, OP y colada.

Controla la pestaña activa y muestra el componente correspondiente.

### Endpoints

- Ninguno directamente; each subvista realiza su propia petición.

### Fallas detectadas

- Estado inicial vista = '' deja la tarjeta sin contenido hasta el primer clic.

## ProgramaAuxiliarBuscaFt.tsx

Busca el **ID de Ficha Técnica** según producto fabricado y alternativa.

Valida inputs numéricos y muestra resultado o mensaje si no existe.

### Endpoints

- **GET** /ficha-tecnica

Query params: productoFabricadoId, alternativa

### Fallas detectadas

- Uso de 999999999 como código de “no existe” es un poco descriptivo.

## ProgramaAuxiliarBuscaOp.tsx

Busca el **ID de Orden de Producción** según la letra de OP formateada.

Aplica `formatearLetraOp` y muestra un spinner durante la petición.

### Endpoints

- **GET** /orden-produccion

Query params: letra

### Fallas detectadas

- Igual que en FT, usa 999999999 para “no existe”.
- Si la API devuelve 0, se interpreta como error aunque podría ser ID válido.

## ProgramaAuxiliarBuscaCol.tsx

Busca el **ID de Colada**, indicando si es de “cintolo”.

Este archivo contiene un desajuste en el nombre del componente.

### Endpoints

- **GET** /colada

Query params: colada, esCintolo

### Fallas detectadas

- El componente se declara como ProgramaAuxiliarBuscaFt en lugar de ProgramaAuxiliarBuscaCol.
- Produce confusión y posibles errores de importación.

## ReglasYVigias.tsx

Archivo con la definición de tipos y lógica para reglas y vigías, **todo comentado**.

No exporta ni renderiza ningún componente activo.

### Endpoints

- Ninguno (código deshabilitado)

### Fallas detectadas

- El código está completamente comentado, no tiene funcionalidad.
- Falta exportación y uso de hooks para configuración o navegación.