

# Diagrama de flujo de navegación y datos

## 🔗 TransformacionesTabla.tsx

Este componente muestra una tabla de órdenes de producción pendientes de transformación.

Se apoya en **OrdenesDeProduccion** para renderizar la UI de la tabla.

### Propiedades

Este componente recibe tres props clave para mostrar las transformaciones.

Propiedad	Tipo	Descripción
pagina	number	Número de página actual.
tablaRef	RefObject<HTMLTableElement>	Referencia al elemento <table>.
transformaciones	ordenDeProduccionType[]	Lista de órdenes para aprobar.

### Renderizado

La función retorna un fragmento JSX con el componente de órdenes.

tsx

```
const TransformacionesTabla: FC<propsType> = ({  
  pagina,  
  tablaRef,  
  transformaciones,  
) => {  
  return (  
    <>  
      <OrdenesDeProduccion  
        columna={'Aprobar'}  
        datosOp={transformaciones}  
        esTransformacion  
        etiquetaBoton={'Aprobar'}  
        lanzarTransformacion={() => {}}  
      />  
    </>  
  );  
};
```

### Integración con OrdenesDeProduccion

Este componente delega la renderización en **OrdenesDeProduccion**.

A continuación se detallan las props que le pasa:

Propiedad	Valor
columna	'Aprobar'
datosOp	transformaciones (array)
esTransformacion	true
etiquetaBoton	'Aprobar'
lanzarTransformacion	función vacía como placeholder

## ModuloPisoDePlanta.tsx

Este módulo controla el acceso y la renderización de rutas del piso de planta.

Verifica permisos de usuario y monta un enrutador secundario.

### Importaciones

- **rutasPisoDePlanta**: definición de ruta base del módulo.
- **useTraerUsuario**: hook personalizado para obtener datos de usuario.
- **EnrutadorSecundario**: componente que monta rutas hijas.

### Hook useTraerUsuario

Este hook devuelve la información del usuario actual.

El objeto `usuario` contiene un array `rutas` con permisos y paths.

### Selección de ruta

Se filtra la ruta que coincide con la del módulo de piso de planta:

ts

```
const ruta = usuario.rutas.find(
  r => r.path === rutasPisoDePlanta.path
);
```

### Renderizado condicional

Si el usuario tiene permiso, se renderiza **EnrutadorSecundario**:

tsx

```
return (
  <>
  {ruta && <EnrutadorSecundario ruta={ruta} />}
  </>
);
```

### Endpoints

- **TransformacionesTabla**: no realiza peticiones HTTP directas.

- **\_ModuloPisoDePlanta**: no realiza peticiones HTTP directas.
- **useTraerUsuario**: probablemente usa GET /api/usuario para obtener datos.

### **Uso del hook usuario**

useTraerUsuario realiza una petición GET a /api/usuario.