

Diagrama de flujo de navegación y datos

Este diagrama muestra la **navegación** entre componentes y el **flujo de datos** asociados a la carga de usuario y renderizado de transformaciones.

TransformacionesTabla.tsx

Este componente presenta una tabla de **transformaciones** de órdenes. Permite aprobar cada transformación mediante un botón integrado.

Props

- **pagina**: número de página actual.
- **tablaRef**: referencia al elemento HTML de la tabla.
- **transformaciones**: lista de registros de tipo `ordenDeProduccionType`.

Renderizado

Renderiza el componente **OrdenesDeProduccion** con configuración especializada para aprobar transformaciones.

Tabla de Props

A continuación se resumen las propiedades en la siguiente tabla.

Prop	Tipo	Descripción
pagina	number	Número de página actual
tablaRef	RefObject<HTMLTableElement>	Referencia al elemento HTML de la tabla
transformaciones	ordenDeProduccionType[]	Lista de órdenes de producción (transformaciones)

Código

Implementación completa del componente.

tsx

```
import { FC, RefObject } from "react";
import { ordenDeProduccionType } from "../../modelos";
import OrdenesDeProduccion from "../../_comunes/ordenesdeproduccion/OrdenesDeProduccion";

type propsType = {
  pagina: number;
  tablaRef: RefObject<HTMLTableElement>;
  transformaciones: ordenDeProduccionType[];
```

```

};

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

export default TransformacionesTabla;

```

Uso de endpoints

- Este componente **no** realiza llamadas directas a **API**.
- Recibe todos los datos de transformaciones a través de la prop **transformaciones**.

Consideraciones

Recomendaciones para el uso del componente.

Importante

TransformacionesTabla no realiza fetch de datos; sus datos provienen de props.

_ModuloPisoDePlanta.tsx

Este componente gestiona el acceso al módulo de **piso de planta**. Carga datos de usuario y despliega rutas secundarias autorizadas.

Dependencias

- **rutasPisoDePlanta**: objeto con configuración de ruta principal.
- **useTraerUsuario**: hook personalizado para obtener datos del usuario.
- **EnrutadorSecundario**: componente que maneja rutas hijas dentro del módulo.

Lógica de renderizado

1. Invoca useTraerUsuario para obtener el objeto usuario.
2. Busca en usuario.rutas la ruta que coincide con rutasPisoDePlanta.path.
3. Si existe, renderiza <EnrutadorSecundario ruta={ruta} />.

Rutas de navegación

- **rutasPisoDePlanta.path**: define el **endpoint** de navegación base del módulo de piso de planta.

Endpoints

- **useTraerUsuario()**: se conecta al endpoint de usuario.
- Método estimado: GET /api/usuario
- Devuelve datos del usuario y sus rutas autorizadas.

Código

A continuación, el código completo del componente.

tsx

```
import { rutasPisoDePlanta } from "../../enrutadores/rutas/rutasPisoDePlanta";
import { useTraerUsuario } from "../../_custom-hooks";
import EnrutadorSecundario from "../../enrutadores/EnrutadorSecundario";

const ModuloPisoDePlanta = () => {
  const usuario = useTraerUsuario();
  const ruta = usuario.rutas.find(r => r.path === rutasPisoDePlanta.path);

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

export default ModuloPisoDePlanta;
```

Consideraciones

Recomendación clave para robustez.

Recomendación

Asegura que useTraerUsuario maneje estados de carga y errores.