BaseComponent<T>

Clase abstracta reutilizable en Angular para facilitar la creación de componentes que usen formularios, tablas, modales y alertas. Provee métodos y propiedades comunes que se pueden extender y personalizar según las necesidades del componente específico.

Objetivo

Esta clase permite centralizar funciones comunes como:

- Inicialización de formularios.
- Configuración y carga de datos en tablas.
- Manejo de paginación.
- Registro de acciones en tablas.
- Control de selección de filas.
- Navegación entre rutas.
- Configuración de modales y alertas visuales.

Inyecciones de Dependencias

- **Router** (`@angular/router`): permite navegar programáticamente entre rutas.
- **FormBuilder** (`@angular/forms`): crea formularios reactivos fácilmente.
- **BreadcrumbService**: establece el breadcrumb de navegación (migas de pan).

Estas dependencias se inyectan usando la función `inject()`.

Propiedades

- **title / subTitle**: Títulos que pueden usarse para encabezados del componente.
- **form**: Instancia del formulario reactivo principal del componente.
- **tableConfig**: Configuración de comportamiento y estilo de la tabla.
- **columns**: Lista de columnas visibles de la tabla.
- **paginationConfig**: Configuración específica para la paginación.
- **dataSource**: Arreglo de datos que alimenta la tabla.
- **page / size**: Controlan la página actual y el tamaño de la paginación.
- **selectedItems**: Elementos seleccionados en la tabla.
- **showMessage**: Bandera que controla si se muestra o no un mensaje.
- **alertOptions**: Configuración para mostrar un mensaje con SweetAlert.
- **modalConfig**: Configuración de un modal genérico.
- **actionHandlers**: Mapa que asocia nombres de acciones con funciones que las manejan.

Métodos Públicos y Protegidos

initForm()

Inicializa el formulario base con un campo `filter`.

```
this.form = this._fb.group({
  filter: ['']
});
```

Es útil para componentes que tengan filtros simples.

setBreadcrumb(items: BreadcrumbItem[])

Configura el breadcrumb que se mostrará en la parte superior del componente, útil para indicar la ruta actual.

```
this._breadcrumbService.setItems(items);
```

initTable()

Inicializa la tabla cargando columnas, configuración y datos. Solo funciona si los métodos getTableColumns(), getTableConfig() y loadDataTable() están implementados en la subclase.

getTableColumns(), getTableConfig(), loadDataTable()

Son métodos opcionales que deben ser sobreescritos por los componentes que extiendan BaseComponent.

```
protected getTableColumns(): TableListColumns[] {
  return [...]; // retorna las columnas
}
```

navigateTo(route: routeParams[])

Permite redirigir a otra ruta usando el Router de Angular.

```
this.navigateTo(['/ruta', id]);
```

actionsTable(event: ITableEmitterAction<T>)

Ejecuta una función asociada a una acción en la tabla. Se debe haber registrado antes usando registerAction().

```
this.registerAction('editar', (item) => this.editarItem(item));
```

handlePaginator(event: IPagination)

Actualiza page y size con los valores del evento y recarga los datos.

From:
https://wiki.adacsc.co/ - **Wiki**

Permanent link:
https://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:new-migracion-sicoferp:front:base-component&rev=1750877880

Last update: **2025/06/25 18:58**