

# Reusable Asset - Dynamic Related List Admin Guide

## Overview

Related lists in Salesforce offer a compact view of a record detail page that **lists** all the items **related** to that record. There are different types of related lists but even the most advanced come with some limitations.

- The maximum number of columns that can be displayed is 10 (enhanced related list)
- Additional clicks are required to apply a filter that is static in nature.
- The sorting ability on records is only available in enhanced related lists
- Lacks pagination. Readability becomes difficult if the number of rows to display is large.
- Inline Editing/Mass editing is not available
- Scrolling through a large set of records could cause performance issues
- Related Parent Object Parent information cannot be shown.
- Who is impacted?
  - Agents are impacted who wish to look up specific information in the related list record. Also, for enterprise customers dealing with multiple business models, not all fields that are relevant to them are exposed while for some not all fields exposed to them are relevant.
- How does this tool help?
  - This reusable tool helps plug in these gaps and much more.
    - No limit to the number of rows that can be displayed both on desktop and phone.
    - Admin can choose a child object name from here which is brought dynamically ensuring the object has a relationship with the parent object.
    - Provide the option to the admin to choose from a server-side driven data full (most recent, true data) or client-side data pull (fast, productive, on-demand pull to refresh stale data) as if your app is operating on a slow network area you might not want to pull all data at once rather should opt for bringing records of first page and pagination happens using server-side actions
    - Ability to choose what fields to be displayed as a user preference without impacting anyone else
    - Option to paginate the recordset with a preset number of records selected for each page
    - Default filter to be applied on the recordset
    - Placeholder to perform mass updates/delete actions on the related list.
    - Gives the ability for admin to choose whether to give end-users the ability to mass delete by showing/hiding a checkbox in the configuration.
    - Also whether edit/delete options should be given or not. We can also choose to hide the filter criteria applied to the table.
    - Ability to search the records in real-time via the search bar. It provides the ability to search on the exposed fields without needing any additional configuration
    - UX consistent with lightning experience on both desktop and phone
    - This component is built completely using HTML tables, thus providing faster performance.
    - Up to second level Parent Record Fields can be shown on the table. For instance here in this table Related Account and Account's owner info is shown.
    - Supports all types of fields including images.

- What is the cost associated with the tool (Performance, heavy custom, maintenance, etc.)
  - The tool utilizes the lightning web component along with apex code. This tool is tested against various network speeds and under conditions mentioned herein.

## Required editions and user permission

- Available in: Lightning Experience
- No special permission needed

## Metadata

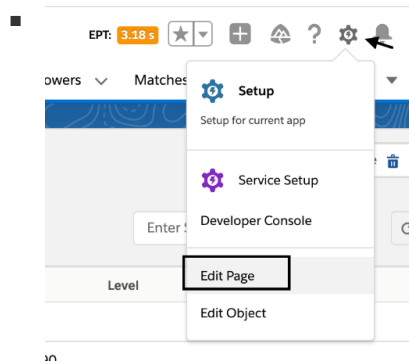
- Component List

S.No.	Name	Type	Description
1	customRelatedList	LWC	Generic Dynamic Lwc component which renders related list for a Parent object and brings data from server real time during pagination, sorting and search events renders as a Tile for small screen size and as a table for medium and large screen sizes
2	generic_RelatedListTable	LWC	Generic Dynamic Lwc component which renders related list for a Parent object and uses client javascript for pagination, sorting and search events renders as a Tile for small screen size and as a table for medium and large screen sizes
3	lazyLoadingtable	LWC	Generic LWC component which loads all records with lazy loading feature bringing data real time from server upon scrolling down
4	multiselect	LWC	Generic Lwc component which gives multiselect dropdown feature in Lightning
5	lightning_utility	LWC	A Utility Lwc component which has commonly used methods for instance make a server call or raise a Toast error message
6	DataTableWrapperController	Apex Class	Apex class containing logic for Search, get records pagination or sorting features for HTML tables
7	ChildObjects	Apex Class	Apex class returning all child objects to choose from for a parent on the config tab for a record page
8	DataTableWrapperController_Test	Apex Class	Test Class for DataTableWrapperController
9	GenericDataUtility	Apex Class	Test data utility class for DataTableWrapperController_Test
10	SLDS_Underscore	Static Resource	Underscore javascript library

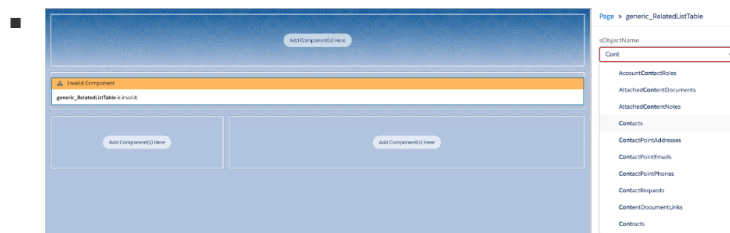
## Initial Setup

- The generic Related List component has been built for lightning\_recordPage to get the child records from the parent id
- Follow the Quick Start here: <https://trailhead.salesforce.com/content/learn/projects/quick-start-lightning-web-components> (Note, install the pre-release plugin if you already have SFDX installed!)
- Clone this repository
- Open your local folder for this repository in VS code.
- Push the source to the desired org

- Follow the below steps to configure a related List for any object
  - Click on the Setup gear icon and Edit page



- Drag and Drop the desired lwc component (generic\_RelatedListTable/customRelatedList) in your record page and select the child subject on the config page



- Give the desired
  - label name for the related section
  - Comma-separated API names of the fields to be displayed
  - filter to be applied joined by AND / OR
  - Default Sort field and Sort direction
  - If Pagination is required select the checkbox “**Pagination Enabled ?**” and input the numeric number for records per page
  - Select the checkbox “**Show row number column ?**” if the row number column is required
  - Select the Checkbox “**Show Checkbox**” if a mass delete/Edit option is required for rows.
  - Select the Checkbox “Hide Search bar” if Search is not required.
  - Select the Checkbox New, Delete, and Edit for respective actions and buttons on the table.
  -

Page > generic\_RelatedListTable

sObjectName

Label to Display

Fields to display

Filters to add

Sort By

Sort Direction

- For Phone Setup - Click on the Edit Record page in Lightning and Click on Activation. You might see the below setup for desktop only

Records per page

■

**ORG DEFAULT** APP DEFAULT APP, RECORD TYPE, AND PROFILE

This page is set as org default for **desktop** form factor.

**i** In standard Salesforce console apps, some objects have a system app default record page. For those objects, if you assign a custom org default page, it doesn't display to users. To enable a custom org default page to show up in the console for those objects, assign a custom page as the app default. [Check your assignments.](#)

- Click on Add another form factor Select phone as an option and save the changes:

- This page is set as org default for **desktop and phone** form factors.

**i** In standard Salesforce console apps, some objects have a system app default record default page, it doesn't display to users. To enable a custom org default page to show page as the app default. [Check your assignments.](#)

## Usage

Post initial Setup users will be able to see the following screen on the Computer screen where the record name is linked to the record page of record :

**Contacts (306)**

Filter Applied : Createddate!=null resulted 306 records

Click on Header section for Sorting records

To Create a new object record → [New](#) [Delete](#)

Enter Search Text...

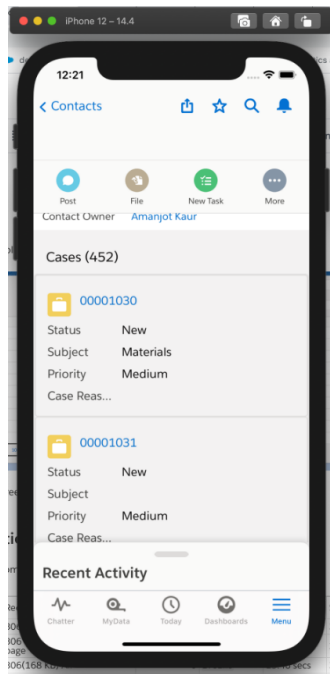
<input type="checkbox"/>	Full Name ↑	Title	Email	Account Name	Full Name	Mobile Phone
<input type="checkbox"/>	Abt. Buchhaltung					
<input type="checkbox"/>	Adrin Gerlich					+49-176-34061490
<input type="checkbox"/>	Amelie von Gleichen					
<input type="checkbox"/>	Amir Batic					+49-152-27816329
<input type="checkbox"/>	Andre Lindhorst		abc@abc.com			+49-172-2039050
<input type="checkbox"/>	Andreas Huber					
<input type="checkbox"/>	Andreas Rock					+43-664-2304152
<input type="checkbox"/>	Andrej Savka					
<input type="checkbox"/>	Andy Young	SVP, Operations	abc@abc.com			(785) 265-5350
<input type="checkbox"/>	Angelique Bertin		abc@abc.com			

Click on number to view records of certain pages

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

[View All](#)


and following view on mobile and small screens where the record name is linked to the record page of record :



1. The view can be modified if users wish to change displayed fields for the tables by clicking on the gear icon and selecting fields to be displayed.

**Contacts (306)** New Delete

Filter Applied : Createddate!=null resulted 306 records

Enter Search 

Select fields to display

4 options selected

- ☒ Last Name
- ☒ First Name
- ☒ Salutation
- ☒ Full Name
- ☐ Other Street

	Last Name	First Name	Full Name ↑	
<input type="checkbox"/>	Buchhaltung	Abt.	Abt. Buchhaltung	
<input type="checkbox"/>	Gerlich	Adrin	Adrin Gerlich	N
<input type="checkbox"/>	von Gleichen	Amelie	Amelie von Gleichen	
<input type="checkbox"/>	Batic	Amir	Amir Batic	N
<input type="checkbox"/>	Lindhorst	Andre	Andre Lindhorst	N
<input type="checkbox"/>	Huber	Andreas	Andreas Huber	N
<input type="checkbox"/>	Rock	Andreas	Andreas Rock	Mr.
<input type="checkbox"/>	Savka	Andrej	Andrej Savka	
<input type="checkbox"/>	Young	Andy	Andy Young	Mr
<input type="checkbox"/>	Bertin	Angelique	Angelique Bertin	Mrs.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

[View All](#)

2. Bulk records can be deleted by selecting the checkbox in the 1st column and selecting the delete button on top.

3. Double click on the row to go on Edit mode and make changes to the row accordingly.

a. **In Progress Cases (448)** Cancel Save New Delete

Filter Applied : Createddate!=null resulted 448 records

Enter 3 Characters or more for Search

	Case Number ↑	Status	Case Origin	Priority	Subject	Description
<input type="checkbox"/>	00001030	New	Phone	Medium	Materials	
<input type="checkbox"/>	00001031	New	Phone	Medium		
<input type="checkbox"/>	00001185	Escalated	Email	Low	testSubject	Neque porro quisquan
<input type="checkbox"/>	00001186	Escalated	Email	Low	testSubject	Neque porro quisquan
<input type="checkbox"/>	00001187	Escalated	Phone	Low	testSubject	Neque porro quisquan

b. For mass edit update the value on any row for a column and select rows on which the same value needs to be applied on multiple rows.

i.

	Case Number ↑	Status	Case Origin	Priority	Subject	Description
<input checked="" type="checkbox"/>	00001030	New	Email	Medium	Materials	
<input checked="" type="checkbox"/>	00001031	New	Phone	Medium		
<input type="checkbox"/>	00001185	Escalated	Email	Low	testSubject	Neque porro quisquan
<input type="checkbox"/>	00001186	Escalated	Email	Low	testSubject	Neque porro quisquan
<input type="checkbox"/>	00001187	Escalated	Phone	Low	testSubject	Neque porro quisquan

4. View all will take users to different screens in a new tab for access to all records with the option to view detail, edit or delete the record.

**Contacts (306)** New

Filter Applied : Createddate!=null resulted 306 records

Enter Search Text..

	<input type="checkbox"/> Last Name	<input type="checkbox"/> First Name	<input type="checkbox"/> Full Name ↑	<input type="checkbox"/> Salutation	
1	<input type="checkbox"/> Buchhaltung	Abt.	Abt. Buchhaltung		
2	<input type="checkbox"/> Gerlich	Adrin	Adrin Gerlich	Mr.	
3	<input type="checkbox"/> von Gleichen	Amelie	Amelie von Gleichen		
4	<input type="checkbox"/> Batic	Amir	Amir Batic	Mr.	
5	<input type="checkbox"/> Lindhorst	Andre	Andre Lindhorst	Mr.	
6	<input type="checkbox"/> Huber	Andreas	Andreas Huber	Mr.	
7	<input type="checkbox"/> Rock	Andreas	Andreas Rock	Mr.	
8	<input type="checkbox"/> Savka	Andrej	Andrej Savka		
9	<input type="checkbox"/> Young	Andy	Andy Young	Mr.	
10	<input type="checkbox"/> Bertin	Angelique	Angelique Bertin	Mrs.	
11	<input type="checkbox"/> Lederer	Anna	Anna Lederer	Ms.	
12	<input type="checkbox"/> Nilsson	Anna	Anna Nilsson		
13	<input type="checkbox"/> Schiek	Anne	Anne Schiek	Miss	
14	<input type="checkbox"/> Hart	Annemieke	Annemieke Hart		
15	<input type="checkbox"/> Huismann	Arie	Arie Huismann	Mr.	
16	<input type="checkbox"/> Cantet	Arnaud	Arnaud Cantet	Mr.	
17	<input type="checkbox"/> Song	Arthur	Arthur Song	Mr.	
18	<input type="checkbox"/> James	Ashley	Ashley James	Ms.	
19	<input type="checkbox"/> Khalil	Ashraf	Ashraf Khalil	Mr.	
20	<input type="checkbox"/> Khalil	Ashraf	Ashraf Khalil	Mr.	
	<input type="checkbox"/> Odevci	Atakan	Atakan Odevci	Mr.	

wascript:void(0)

Show details  
Edit  
Delete

## Performance Analysis

- Below observation was seen when the component was rendered in slow and fast network speed

Component type	Pagination type	Records retrieved	Pagination	Apex call duration	Seconds	Speed
generic_RelatedListTable	Client Side	306(168 KB) All	0	661ms	2.54secs	30mbps
generic_RelatedListTable	Client Side	306 records (168 kB) 10 per page	10	649 ms	2.47secs	30mbps
generic_RelatedListTable	Client Side	306(168 KB) All	0	1781ms	13.46 secs	376 kb/s
generic_RelatedListTable	Client Side	306 records (168 kB) 10 per page	10	1681ms	9.54 secs	376 kb/s
customRelatedList	Server Side	306(168 KB) All	0	901ms	3 secs	30mbps
customRelatedList	Server Side	306 records (168 kB) 10 per page	10	572ms	2.79secs	30mbps
customRelatedList	Server Side	306(168 KB) All	0	1489ms	13.46 secs	376 kb/s
customRelatedList	Server Side	306 records (168 kB) 10 per page	10	1117ms	11.9 secs	376 kb/s

## Limitations

- While using the server-side LWC component (**customRelatedList**) please consider
  - the maximum offset limit is 2000 for Pagination (salesforce imposed by default on SOQL query )
  - Also, the sorting ability on fields of data type 'text area' is disabled by design considering Salesforce SOQL doesn't allow filter and Sorting on "text area" fields.
  - works best if you are dealing with a huge set of data, and the application is rendering on a slow network thus all data shouldn't be brought in one go
- While using the client-side LWC component (generic\_RelatedListTable) please consider using
  - if operating in a high network area as all actions are client-side and provide master performance as compared to server-side.
  - No limit on offset for pagination.
  - can sort on Text area fields as well.

## Error Handling In the component

- lightning\_utility lwc component has Generic methods for Server call which makes the call to apex asynchronously and process error.
  - The component needs to import the lightning\_utility component.

```
import { LightningElement } from 'lightning/uiRecordApi';  
import utility from 'c/lightning_utility';  
import { ShowToastEvent } from 'lightning/platformShowToast';
```

- Every server call is made via the method in the Utility component.

```
utility._servercall(  
    searchrecords,  
    params,  
    this.handleSuccesssearch.bind(this),  
    this.handleError.bind(this)  
);
```

- Response from Server is always received in the form of Lightning\_RemoteResponse wrapper class where payload attribute (type object ) has the response from the AuraEnabled methods.
- Thus Server call method of the lightning\_utility component process the response from the apex by processing the payload attribute in case of exception and raise the toast event.



```

const _servercall = (_serveraction, _params, _onsuccess, _onerror) => {
  if (!_params) {
    _params = {};
  }
  _serveraction(_params)
    .then(_result => {
      if (_result && _onsuccess) {
        if (_result.message == 'Success') {
          _onsuccess(_isSuccessObject(_result));
        } else if (_result.message == 'Fail' || !_result.isSuccess) {
          _onerror(_reduceErrors(_result.payload));
        }
      }
    })
    .catch(_error => {
      if (_error && _onerror) {
        _onerror(_reduceErrors(_error));
      }
    });
};

const _isSuccessObject = (_result) => {
  if (_result.payload != undefined && _result.message == 'Success') {
    var payload = JSON.parse(_result.payload);
    if (payload.lstDataTableRecs != undefined) {
      payload.lstDataTableRecs;
      return payload.lstDataTableRecs;
    }
  } else if (_result.message == 'Fail' || !_result.isSuccess) {
    return undefined;
  }
};

```

## Github Link

<https://git.soma.salesforce.com/amanjot-kaur/lwc-reusable-assets>