Lightning Forms

1 — Last update: 2018/04/10

Lightning Tools

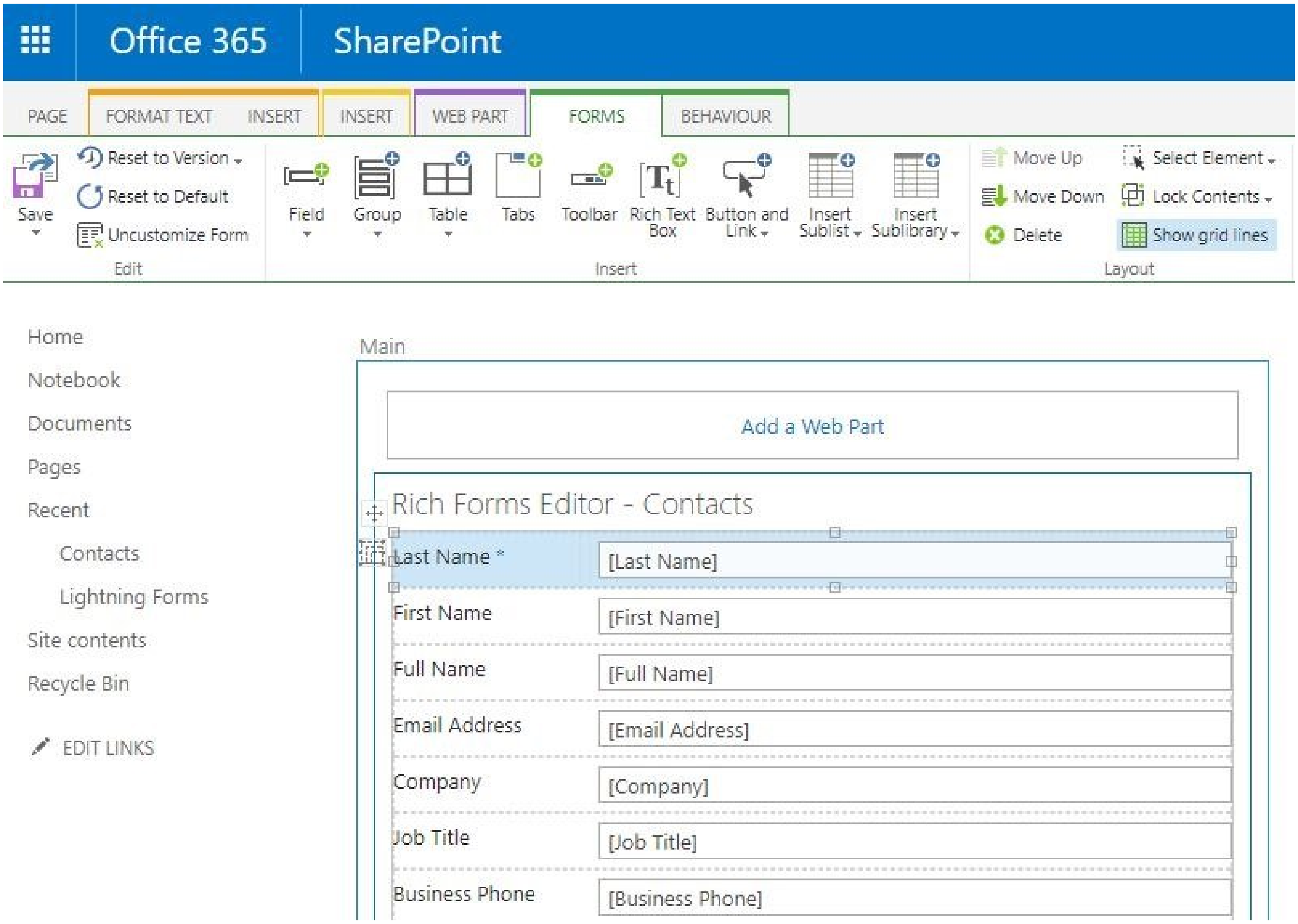
Table of Contents

1. Introduction

Lightning Forms

Lightning Forms is an app for SharePoint that provides an easy to use, yet powerful editor to customize list forms.

Building forms in SharePoint and Office 365 that look great and fulfil all the different purposes of your forms has never been easier!



Last Name * [Last Name]

First Name [First Name]

Full Name [Full Name]

Email Address [Email Address]

Company [Company]

Job Title [Job Title]

Business Phone [Business Phone]

Deploy easily

Lightning Forms is a SharePoint app that extends SharePoint's content editor. No need to use any separate tools or go to an external site. All form editing can be done seamlessly in your browser on your SharePoint site.

Place form fields freely

Place your fields anywhere on the form using easy drag-and-drop or ribbon interface. Start out with the form layout as provided by SharePoint and then reposition fields, align field labels or change the label text where needed.

Group vertically, horizontally or in tabs

Easily group fields and other elements in vertical/horizontal groups or tabs. Flexibly combine different group types to create the layout you want. You can even embed groups inside other groups, allowing for even the most complex layouts.

Rich text editing

Use all the powerful features of SharePoint's content editor web part to add freely formatted text, tables, images, embedded scripts and media using the familiar Office ribbon-interface. And if the user-friendly editor is not enough for you, you can always resort to editing the HTML source to make almost anything a browser can handle possible.

Edit sub items in a grid

Place a sub list on your form for any related SharePoint list. Thus users can fill out multiple entries quickly with SharePoint's familiar editable grid view. Lightning Forms will ensure that the items are saved with the form's other data. And it works even on new forms, so your users can fill out forms quickly even if multiple sub items need to be entered.

Handle complex business logic

Implement even complex business logic using Lightning Forms' flexibly configurable, expression-based behaviours. Make a field or form section dynamically show or hide or become enabled or disabled depending on other inputs. Set default values and even recalculate them while the user is entering data in the form.

Act on data

There is more to data then editing and storing. Users want more than just viewing data they entered. Lightning Forms let's users act on data, making data do more. Build completely new functionality using the freely configurable actions provided in Lightning Forms. Sending mail, adding or changing other list items, redirecting to a web page and much more can easily be done and can even be based on form data using highly flexible expressions.

Define complex validation rules easily

Use expression-based validation rules to specify required form input. These can even handle changes to entered data. For example: Make the entry of a reason required depending on the status selected.

Embed any web part on your form

Using Lightning Forms you can embed any web part inside your form. For example display all related lists and libraries right inside your display form.

Revert to older versions

Whenever you need to go back to an older version of your form, rest assured Lightning Forms editor will automatically keep old versions and let you restore them easily.

See the result while designing

Being able to see what the result will look like makes creating beautiful forms easy. No need to configure where you want your fields in an abstract editor. Immediately get feedback on how things look using the powerful drag-and-drop and ribbon interface provided by Lightning Forms WYSWYG editor.

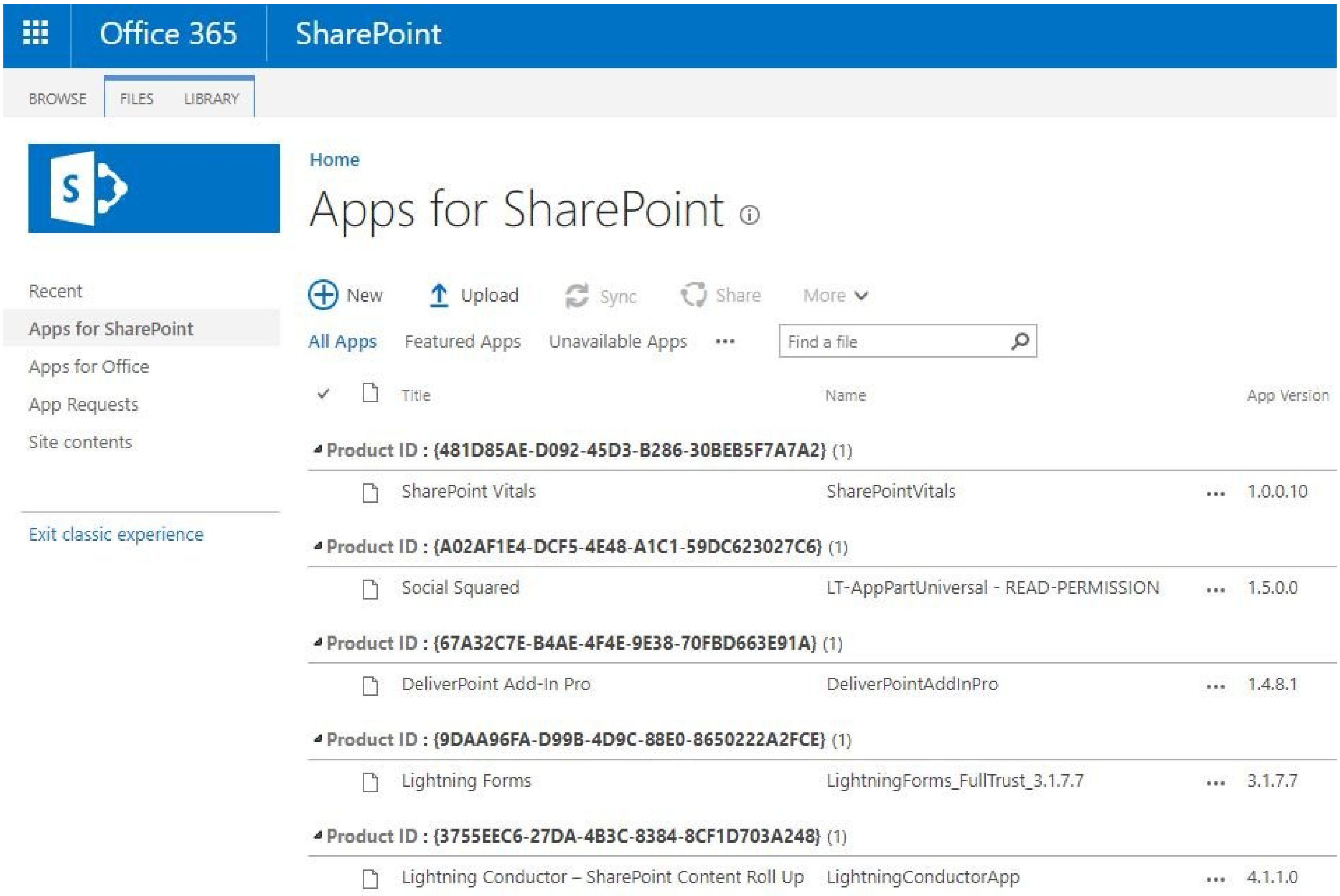
Seamlessly integrated in SharePoint

Because Lightning Forms only extends SharePoint's list forms and uses all out-of-the-box functionalities, the forms work and act as part of SharePoint and users are immediately familiar with them.

1.1. Deploying Lightning Forms

Lightning Forms can be deployed to SharePoint On-Premises (2013 onwards) or SharePoint Online via the App Catalog. Once the App has been made available in the App Catalog, users can deploy the App within their SharePoint Team Sites, and begin to customize the forms.

> ✻ To familiarize yourself with the SharePoint App Catalog, please refer to the steps indicated by Microsoft here.

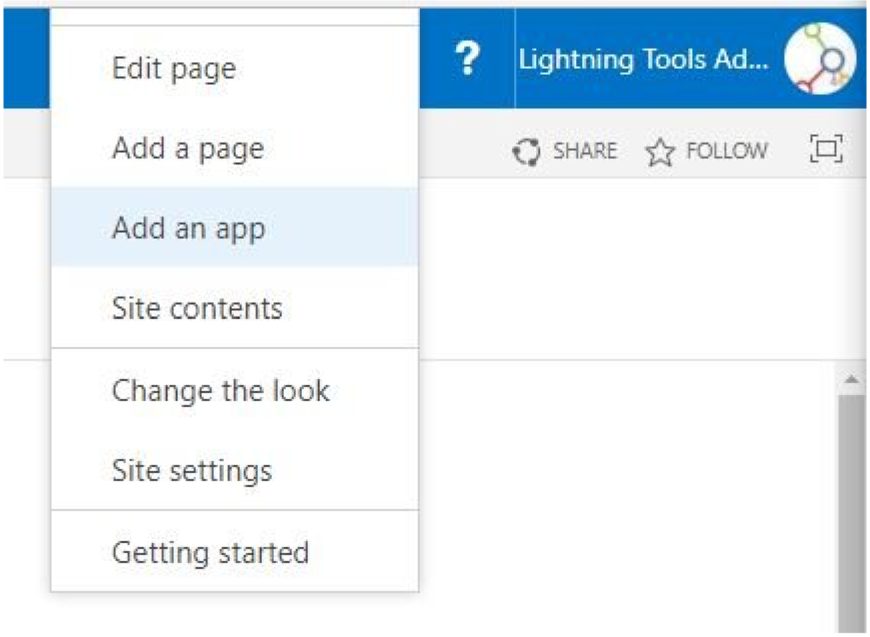


Title	Name	App Version
Product ID : {481D85AE-D092-45D3-B286-30BEB5F7A7A2} (1)		
SharePoint Vitals	SharePointVitals	1.0.0.10
Product ID : {A02AF1E4-DCF5-4E48-A1C1-59DC623027C6} (1)		
Social Squared	LT-AppPartUniversal - READ-PERMISSION	1.5.0.0
Product ID : {67A32C7E-B4AE-4F4E-9E38-70FBD663E91A} (1)		
DeliverPoint Add-In Pro	DeliverPointAddInPro	1.4.8.1
Product ID : {9DAA96FA-D99B-4D9C-88E0-8650222A2FCE} (1)		
Lightning Forms	LightningForms_FullTrust_3.1.7.7	3.1.7.7
Product ID : {3755EEC6-27DA-4B3C-8384-8CF1D703A248} (1)		
Lightning Conductor – SharePoint Content Roll Up	LightningConductorApp	4.1.1.0

Adding Lightning Forms to a SharePoint Site

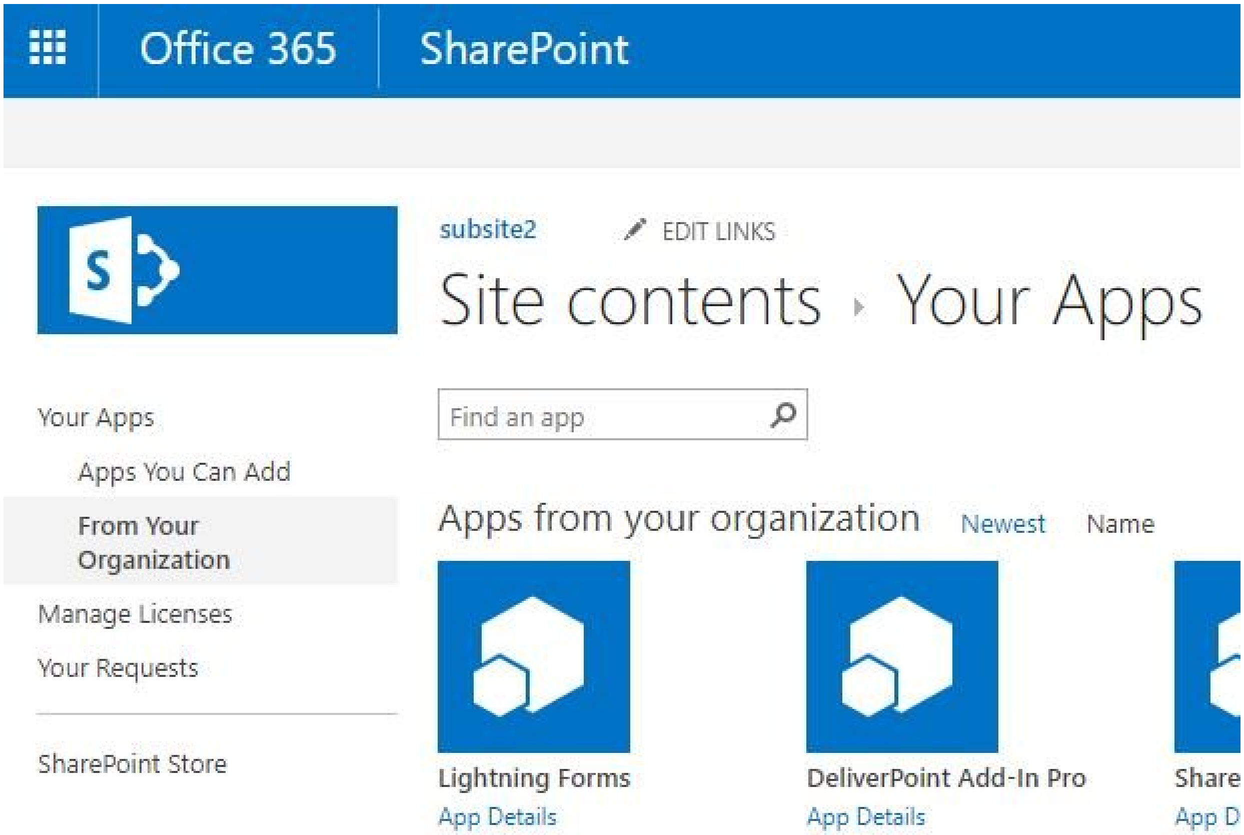
Once the Lightning Forms app is added to the SharePoint App Catalog, Users will be able to add Lightning Forms to their sites following these steps:

1. Navigate to the site where you would like to add Lightning Forms.
2. Click the Site Actions Cog in the top right hand corner of your site, and then choose **Add an app**.



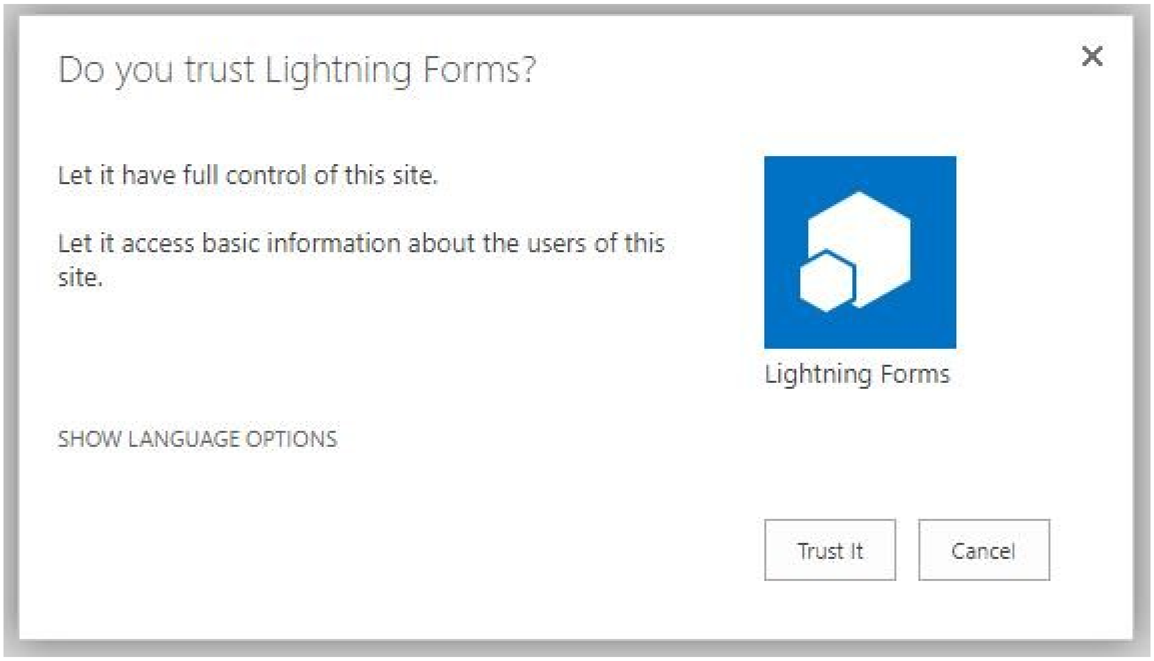
Add an app

1. From the left hand navigation pane under Your Apps, click **From Your Organization**.
2. Click the Lightning Forms tile to add Lightning Forms



Adding Lightning Forms

1. When prompted, choose **Trust It** from the Do you trust Lightning Forms? dialog.



Do you trust Lightning Forms?

Let it have full control of this site.

Let it access basic information about the users of this site.

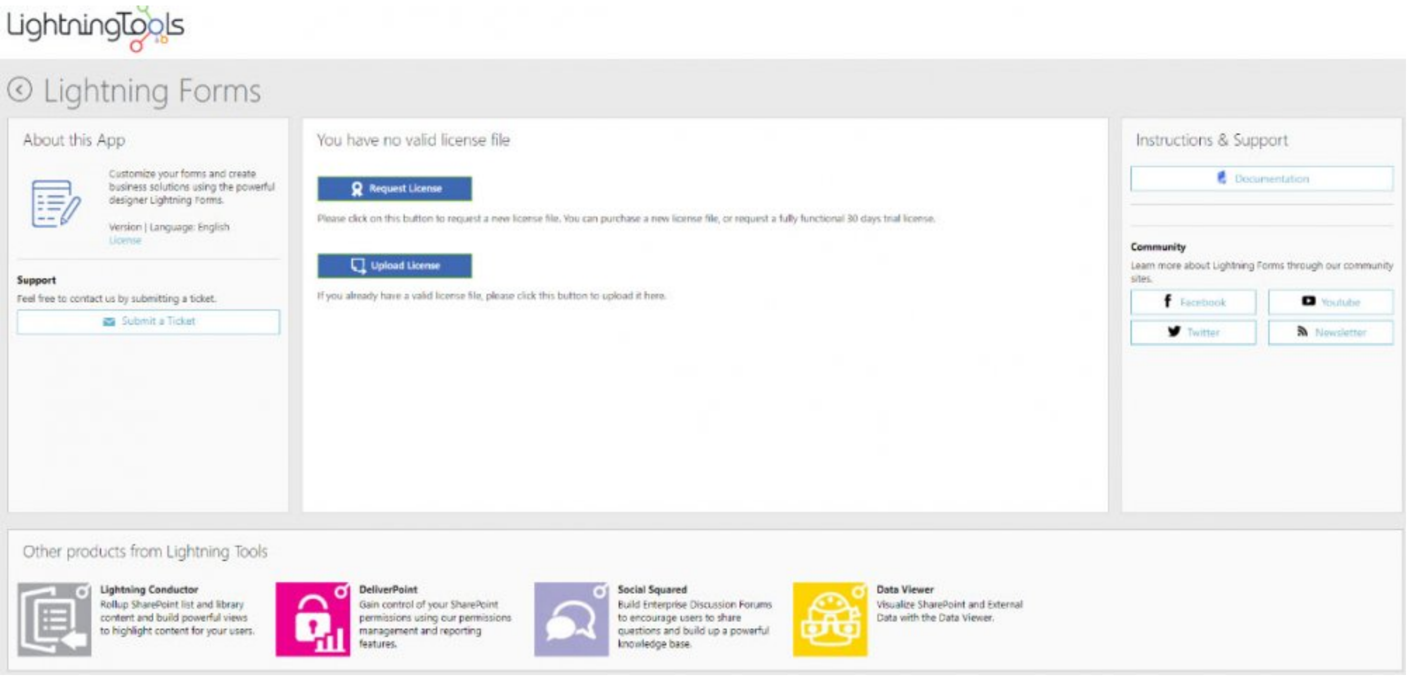
Trust Lightning Forms

Lightning Forms will now successfully be added to your SharePoint site.

<< 1. Introduction
1.2. Licensing >>

1.2. Licensing

After installing Lightning Forms it is required to have a valid license file. A license file can be requested in order to trial Lightning Forms. This file can be requested by the license request form on the app homepage.



Customize your forms and create business solutions using the powerful designer Lightning Forms.

Version | Language: English
License

Support
Feel free to contact us by submitting a ticket.

Please click on this button to request a new license file. You can purchase a new license file, or request a fully functional 30 days trial license.

If you already have a valid license file, please click this button to upload it here.

Community
Learn more about Lightning Forms through our community sites.

Lightning Conductor
Rollup SharePoint list and library content and build powerful views to highlight content for your users.

DeliverPoint
Gain control of your SharePoint permissions using our permissions management and reporting features.

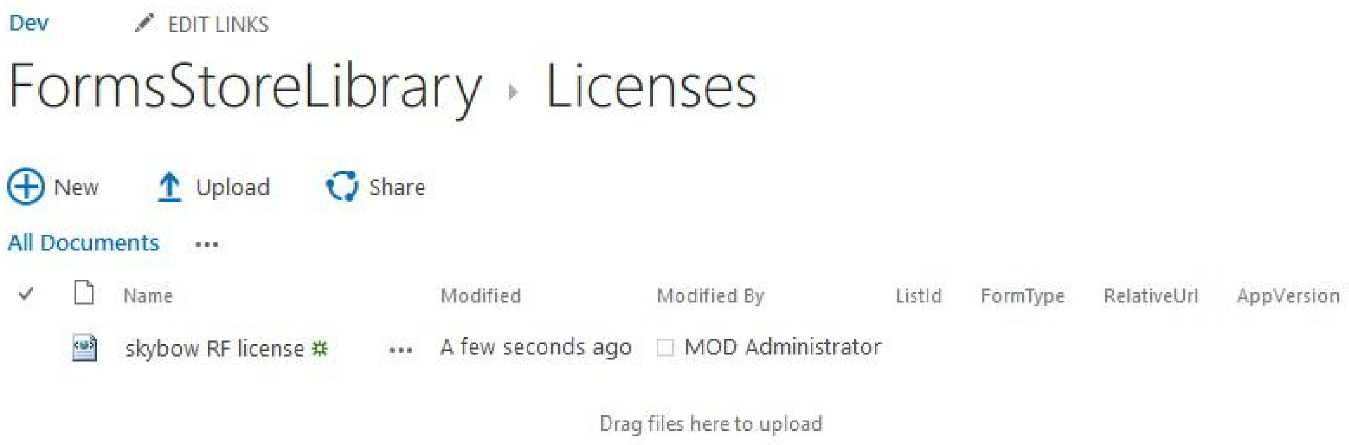
Social Squared
Build Enterprise Discussion Forums to encourage users to share questions and build up a powerful knowledge base.

Data Viewer
Visualize SharePoint and External Data with the Data Viewer.

By clicking on Request License a request form will open where you have to fill in some information and click send.

After requesting the license you receive a license file from us that you can upload by clicking on upload license.

The license is placed into the hidden document library called FormsStoreLibrary in the folder Licenses

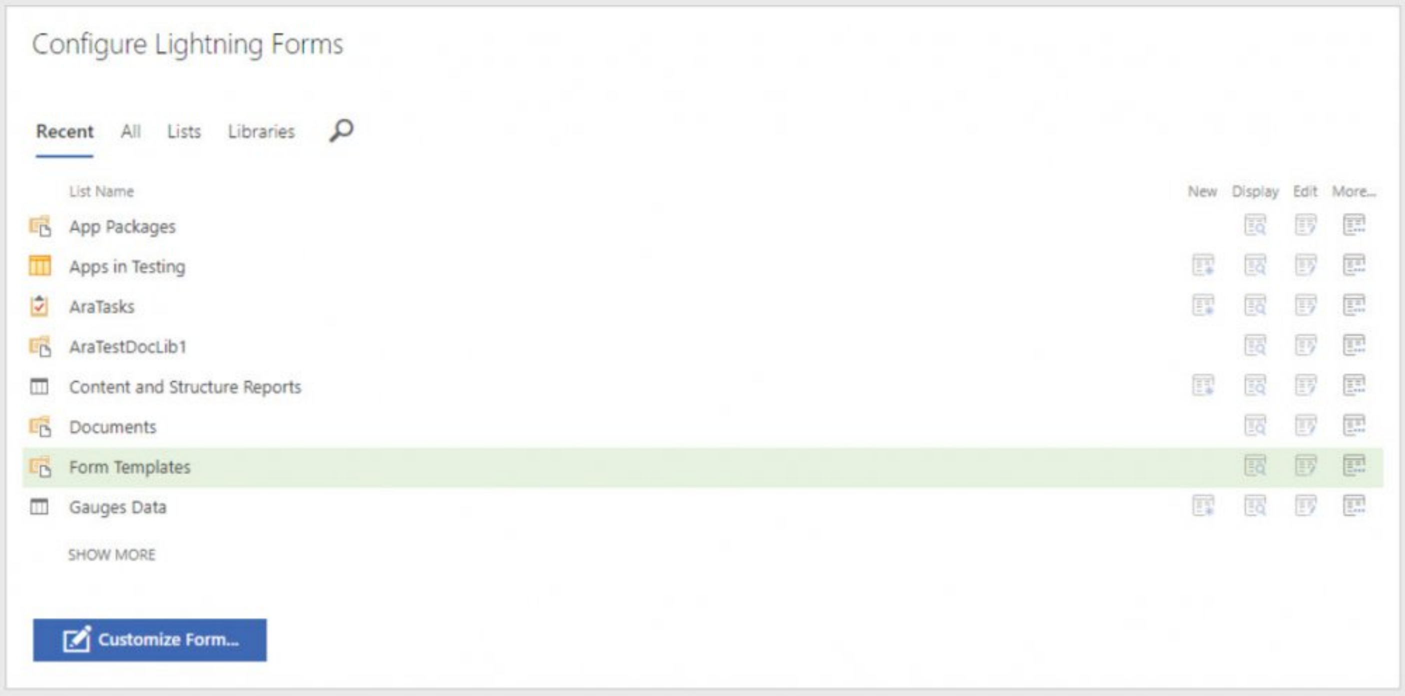


✓		Name		Modified	Modified By	ListId	FormType	RelativeUrl	AppVersion
		skybow RF license ✲	...	A few seconds ago	□ MOD Administrator				

If you have an error during uploading named: “Error occured while uploading the file. Please contact our support.” then the normal issue is that the filename of the license is too long. Just shorten the filename and upload it again.

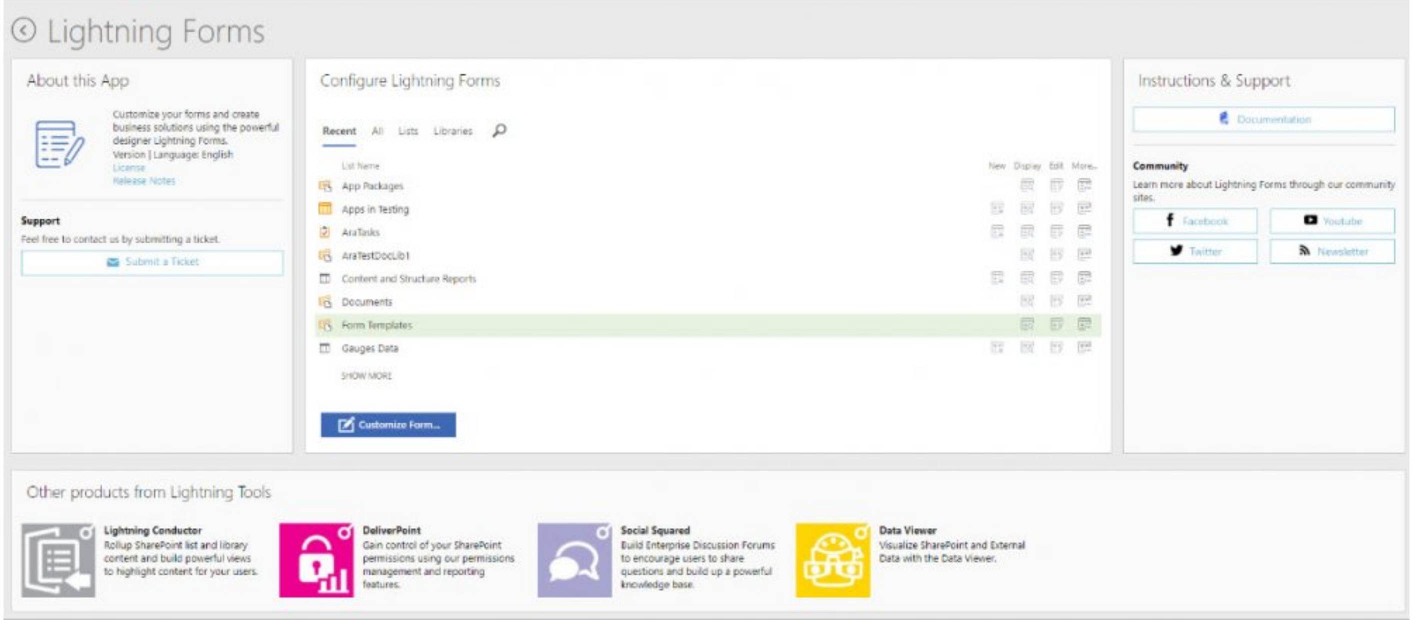
1.3. App Home Page

1. Forms List
In the list overview you can directly customize a form by selecting the appropriate form’s icon under the New, Display and Edit headings. Alternatively the Select Form to Customize button provides a window to select the list and form to customize. Select Recent, All, Lists and Libraries to change which lists and libraries are shown or use the Find a list search box to filter the lists displayed.


Configure Lightning Forms

List Name	New	Display	Edit	More...
App Packages		▤	▤	▤
Apps in Testing	▤	▤	▤	▤
AraTasks	▤	▤	▤	▤
AraTestDocLib1		▤	▤	▤
Content and Structure Reports	▤	▤	▤	▤
Documents		▤	▤	▤
Form Templates		▤	▤	▤
Gauges Data	▤	▤	▤	▤

From the Application Page, you can Submit a Support Ticket to our Support Portal, Link to the documentation, join our Social Community, and view other accelerators from Lightning Tools.



Lightning Forms

About this App

Customize your forms and create business solutions using the powerful designer Lightning Forms.
Version | Language: English
License
Release Notes

Support

Feel free to contact us by submitting a ticket.

Submit a Ticket

Configure Lightning Forms

Recent All Lists Libraries

List Name	New	Display	Edit	More...
App Packages		▤	▤	▤
Apps in Testing	▤	▤	▤	▤
AraTasks	▤	▤	▤	▤
AraTestDocLib1		▤	▤	▤
Content and Structure Reports	▤	▤	▤	▤
Documents		▤	▤	▤
Form Templates		▤	▤	▤
Gauges Data	▤	▤	▤	▤

SHOW MORE

Customize Form...

Instructions & Support

Documentation

Community

Learn more about Lightning Forms through our community sites.

Facebook Youtube
Twitter Newsletter

Other products from Lightning Tools

Lightning Conductor
Rollup SharePoint list and library content and build powerful views to highlight content for your users.

DeliverPoint
Gain control of your SharePoint permissions using our permissions management and reporting features.

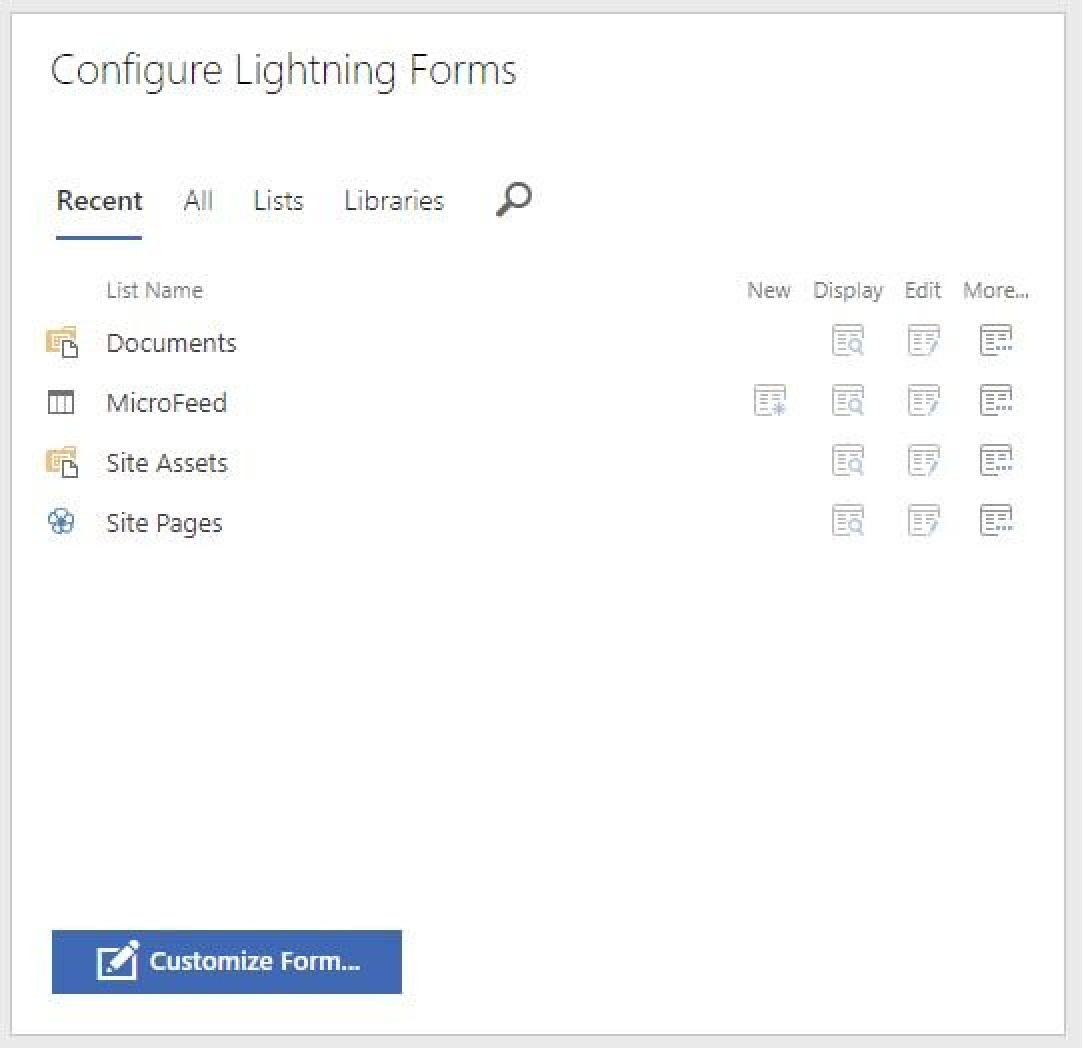
Social Squared
Build Enterprise Discussion Forums to encourage users to share questions and build up a powerful knowledge base.

Data Viewer
Visualize SharePoint and External Data with the Data Viewer.

1.4. Start Customizing Form

You can start customizing a form by opening it in design mode process in 3 ways:

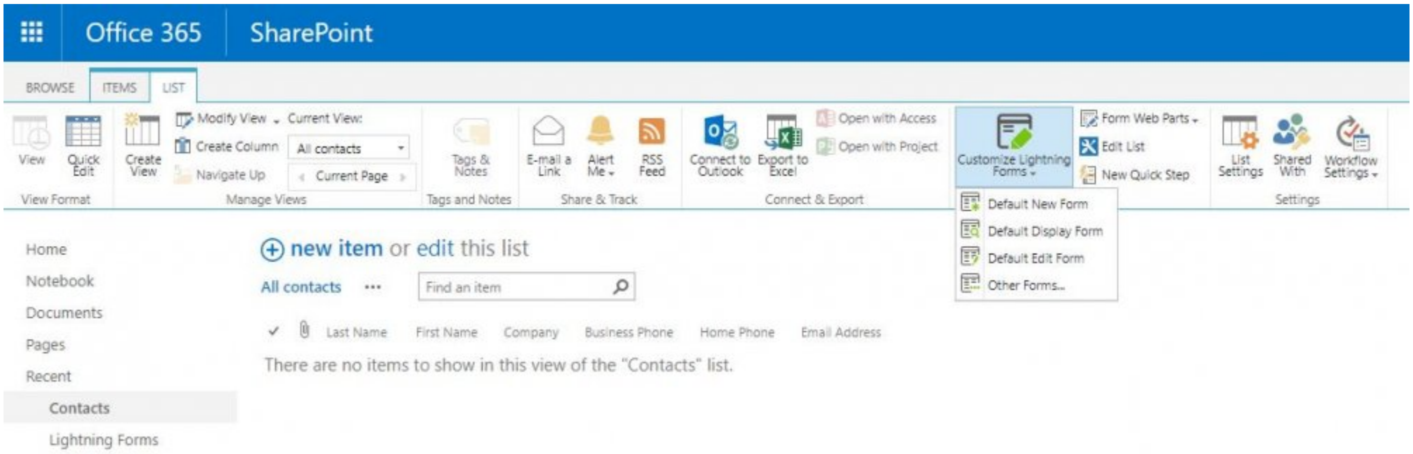
1. From the Main App Page



Specify a list and select the form you want to customize. For more information, see Main App Page.

2. From the List View Page

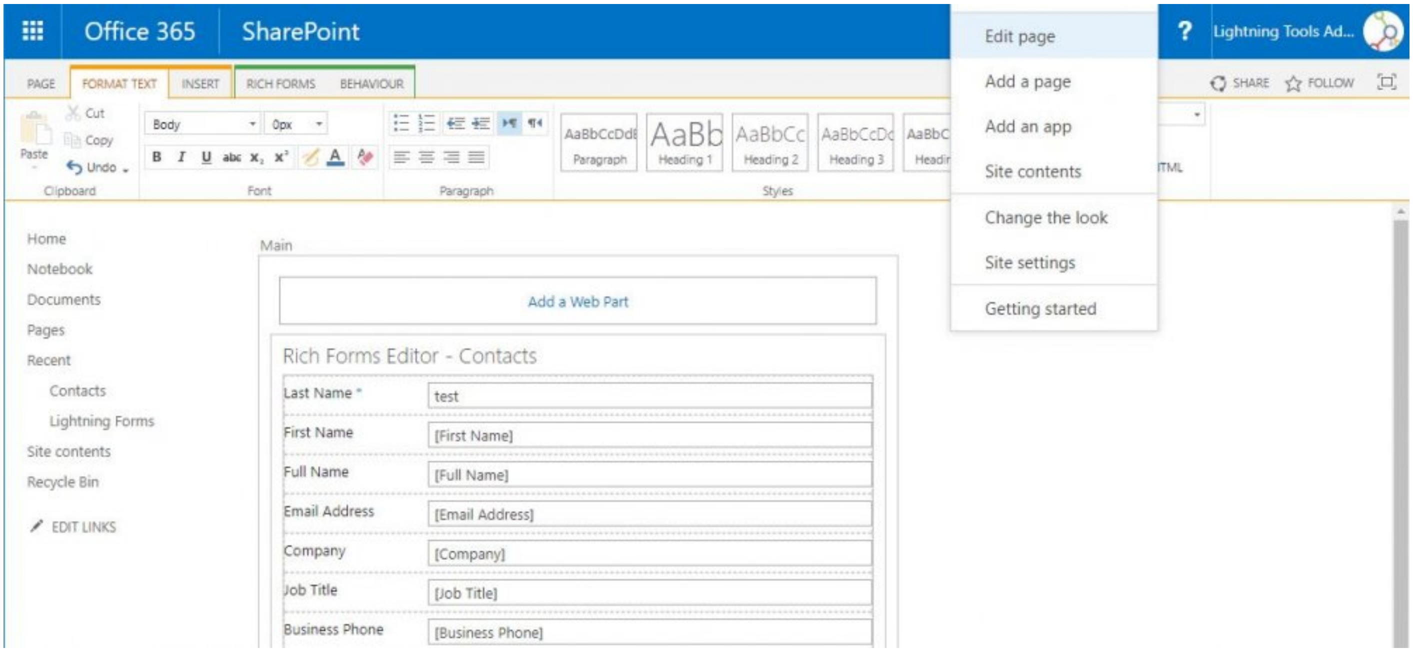
Select the Customize Rich Forms option on the List View ribbon.


Office 365
SharePoint
BROWSE
ITEMS
LIST
Customize Lightning Forms
Default New Form
Default Display Form
Default Edit Form
Other Forms...
new item or edit this list
All contacts
There are no items to show in this view of the "Contacts" list.

Customizing Lightning Forms

3. From the Form Page

Click Edit Page action in the Settings menu of the form page.



Note: This will only allow further customizing the form using Lightning Forms features if the form has previously been customized by one of the other above ways.

After you perform one of the above actions, the form is opened in design mode and you can start customization using the Lightning Forms Editor features:

Field
Group
Tabs
Toolbar
Rich Text Box

[<< 1.3. App Home Page]
[2. Form Editor >>]

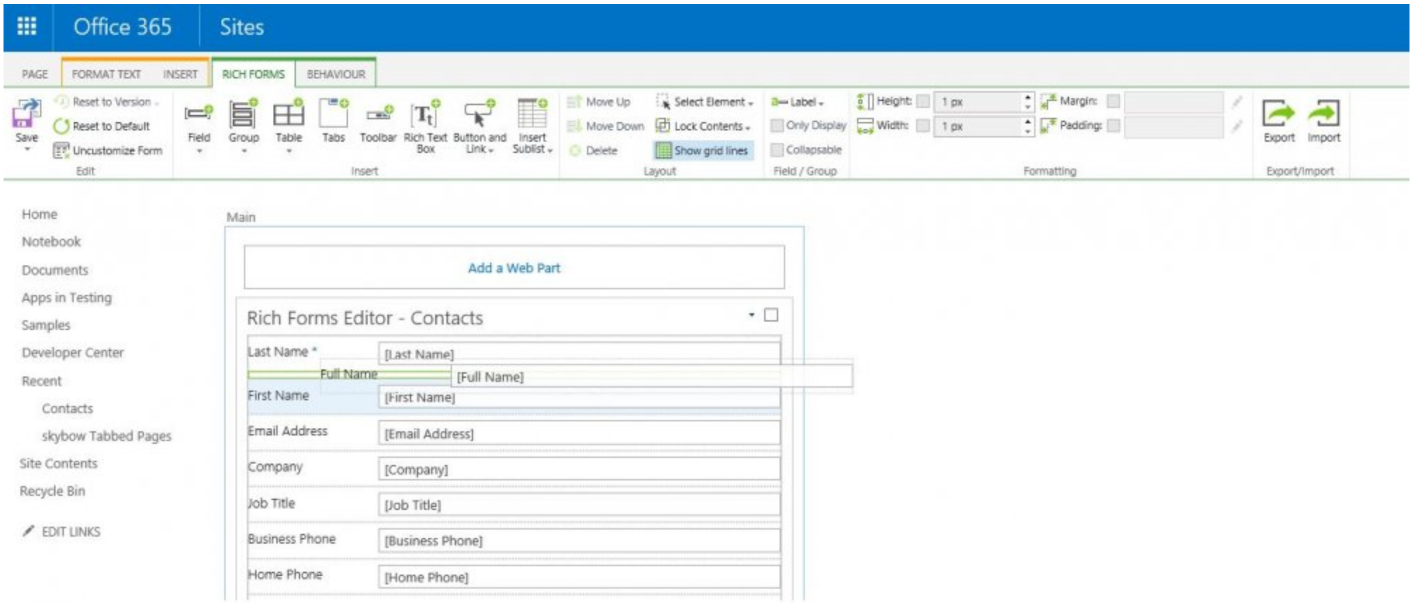
2. Form Editor

The main form editor features are organized into a series of tabs on the editor ribbon. Once you open your form in design mode, 2 additional tabs will appear after standard SharePoint tabs.

- Controls for adding, editing, and customizing content are prominently displayed on Rich Forms tab.
- Expression-based behaviours for configuring complex business logic are available on Behaviour tab.

Drag and Drop

The Form Editor uses a drag and drop interface. Items can be dragged and dropped to reposition them. When dragging an item, move it over the edge of an existing item. A border line will appear indicating the position where it will be inserted. You can drag and drop one item, as well as a group with multiple items in it.

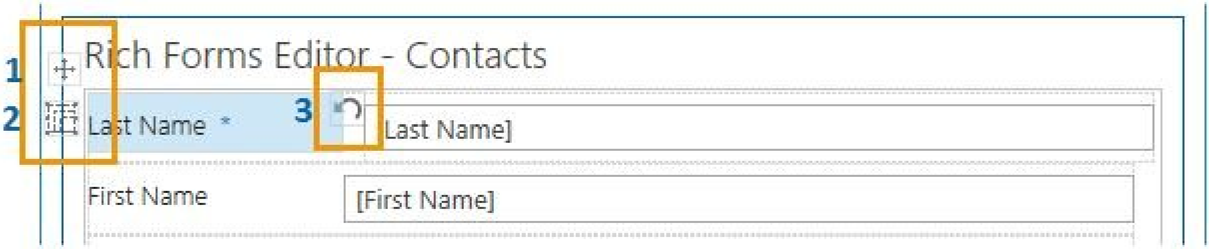


Move your mouse over an item on your form. When the item becomes highlighted, press your left mouse button and drag the placeholder over an existing form item. Position the item so that border line appear to indicate that the position is valid. Release the mouse button.

For already selected items there is a Move selected element icon at the left side. When you click on it the cursor changes to the drop one until you release the left mouse button.

Grippers

Grippers are those small icons that appear when you select an element inside the Forms Editor.

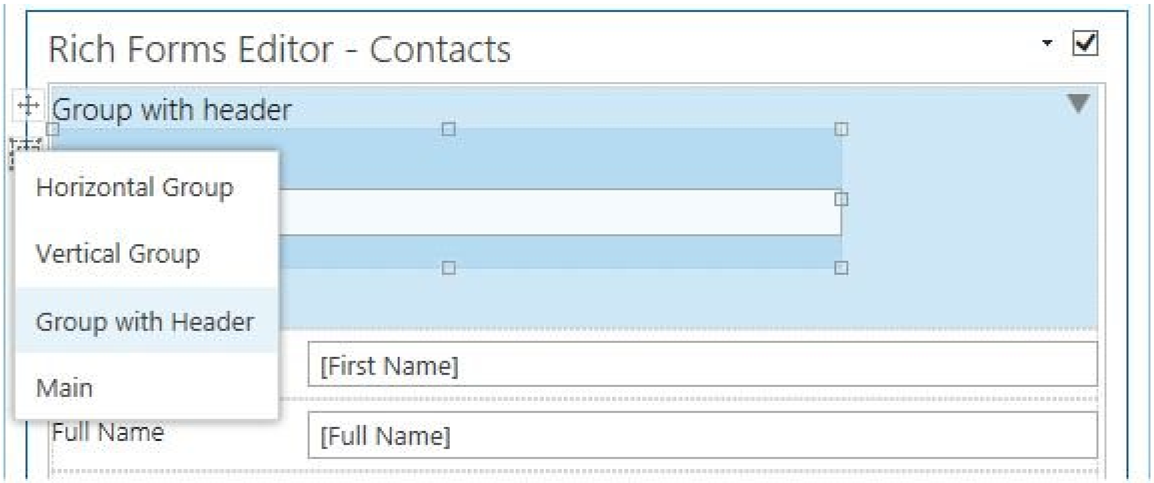


1. Move selected element

Is used to move items: items can be dragged and dropped to reposition them. When you click on it the cursor changes to the drop one until you release the left mouse button.

2. Select surrounding group

Opens context menu with surrounding groups and allows to select the needed element without using a Select Element button on the Rich Forms ribbon.



Group with Header

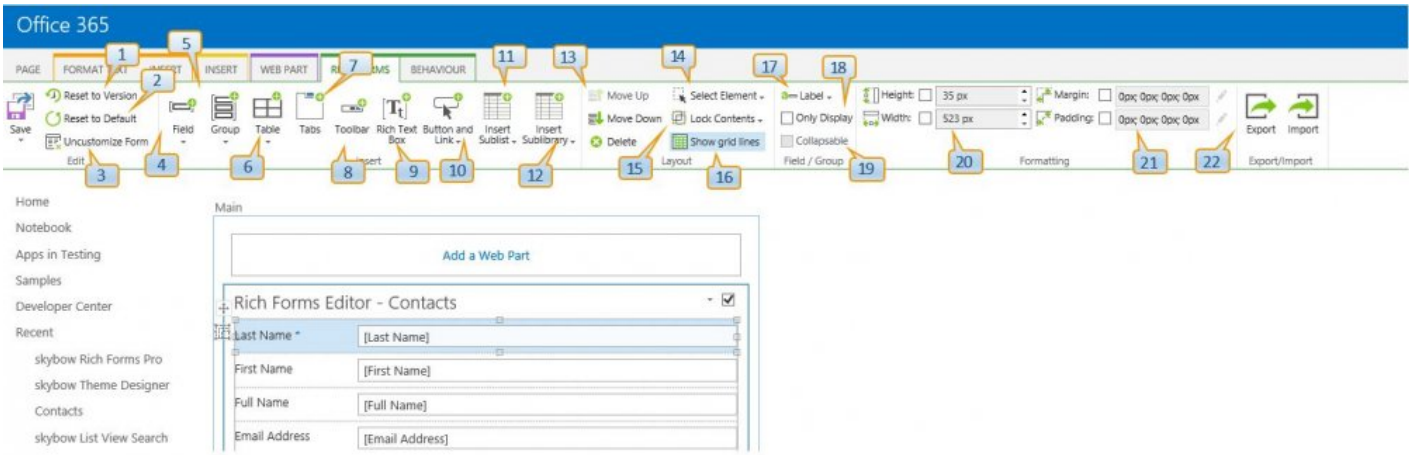
3. Revert to default label

Gripper that becomes available when label is selected. If some changes were performed on the label, clicking on it will revert to the default label state.

2.1. Rich Forms Tab

Rich Forms Ribbon Tab Features

Rich Forms tab contains controls for adding, editing, and customizing content of the form. Once you open your form in design mode, it will appear after standard SharePoint tabs.



Rich Forms Tab

1. Reset to Version

Reset to Version action provides the ability to reset the form content to any saved version.

Whenever you need to go back to an older version of your form, rest assured Lightning Forms editor will automatically keep old versions and let you restore them easily.

2. Reset to Default

Reset Form action resets the form content to the initial form template.

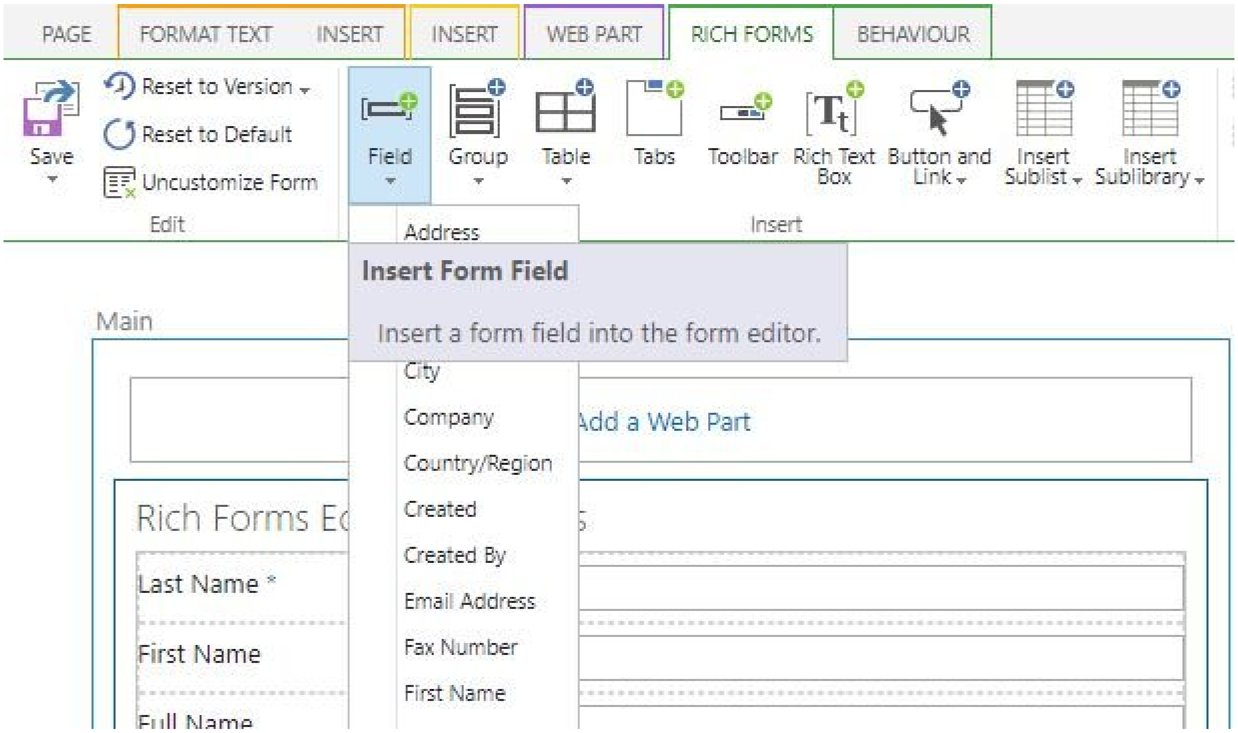
3. Uncustomize Form

Uncustomize Form action removes Lightning Forms customization of the current form and reverts to the standard SharePoint form.

4. Field

Field is a custom action in the ribbon toolbar to add fields into the content editor web part. It allows to add any field available for the current form.

It also provides an ability to add new fields directly from the current page: clicking Add New button initiates opening of Create Column dialog. New field is immediately added to the form.



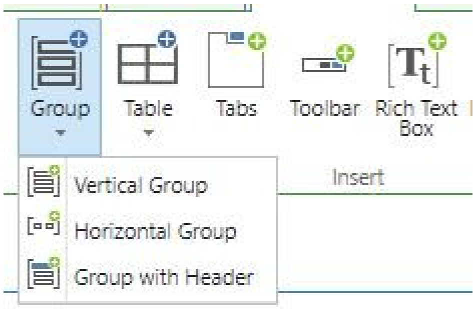
Add a Field

5. Group

Group is a custom action in the ribbon toolbar to add groups into the content editor web part.

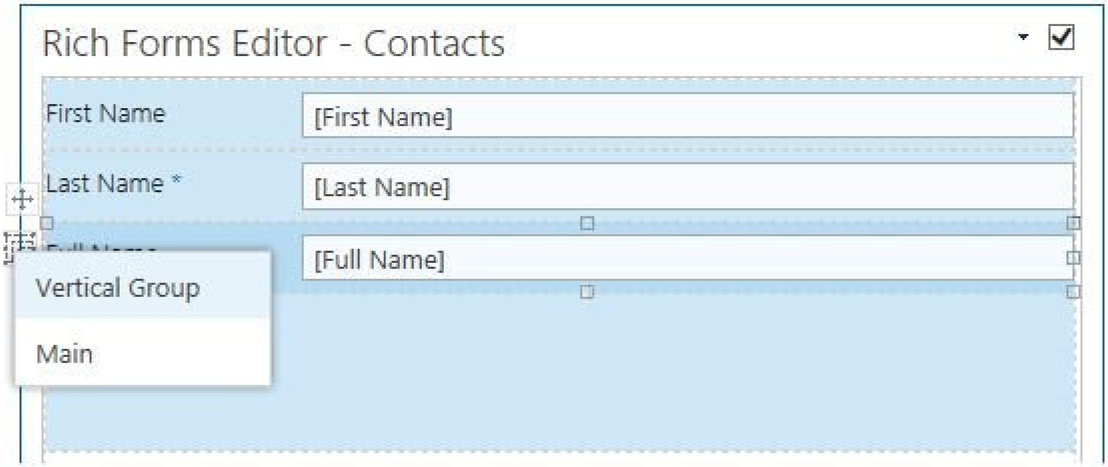
Since positioning of fields is essential for designing your forms to match the structure of the data represented, Lightning Forms provides Groups functionality that allows to manipulate form field positions, flexibly combine different group types to create the layout you want. You can even embed groups inside other groups, allowing for even the most complex layouts.

There are 3 available group types: vertical, horizontal and group with header.



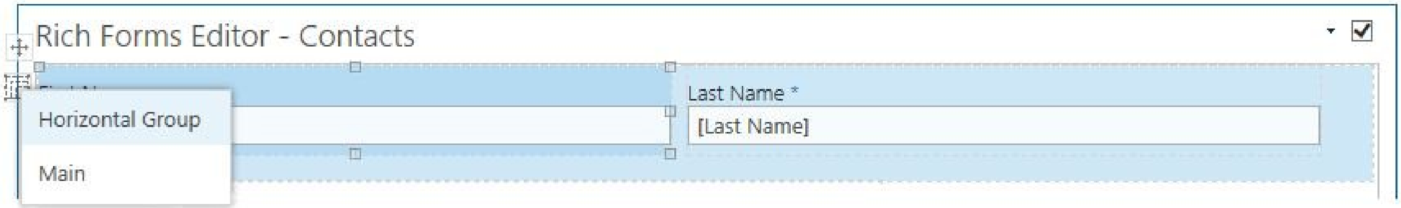
Lightning Forms Group

Vertical group allows to place form fields in vertical stack one below another.

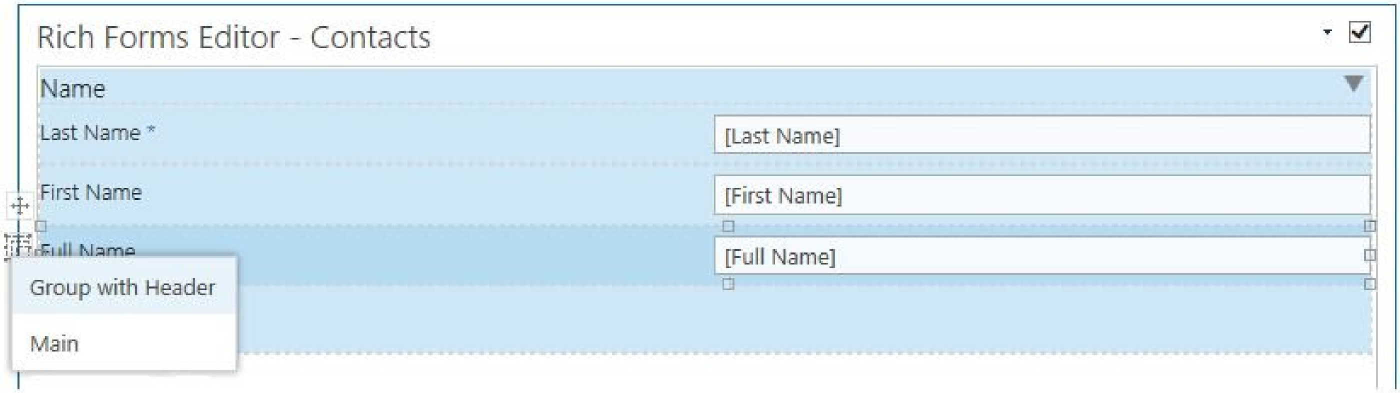


Vertical Group

Horizontal group allows to place form fields in horizontal stack one after another.



Group with header allows to place form fields in vertical stack one below another and to display a header for it. Group with header follows the layout rules of vertical group. There is a setting on the ribbon to define if it will be collapsible: **Collapsible** checkbox allows to specify if an icon should appear to collapse/expand the content of this group.

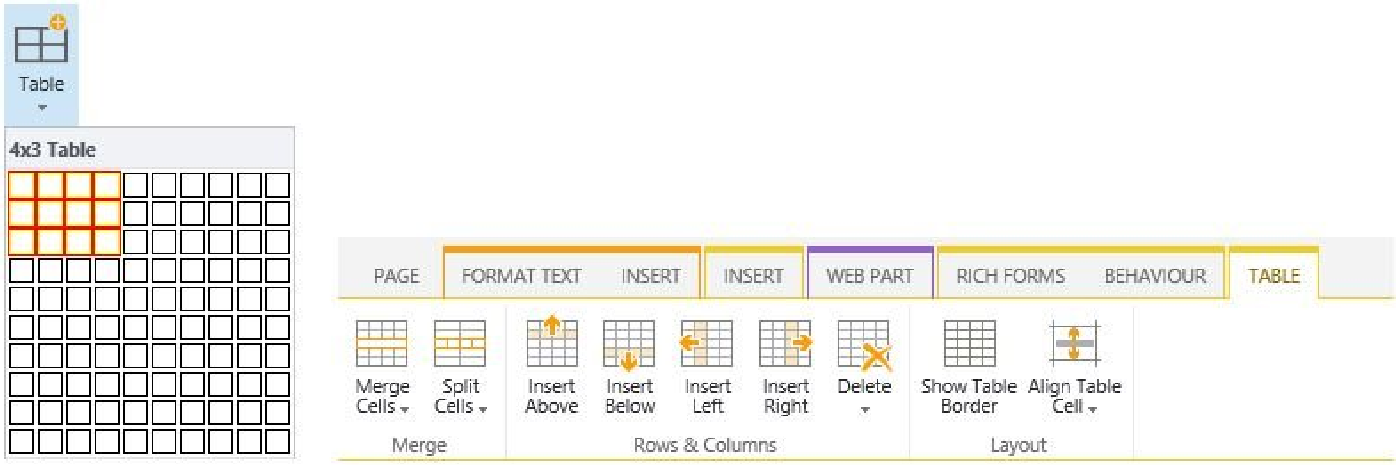


Group with Header

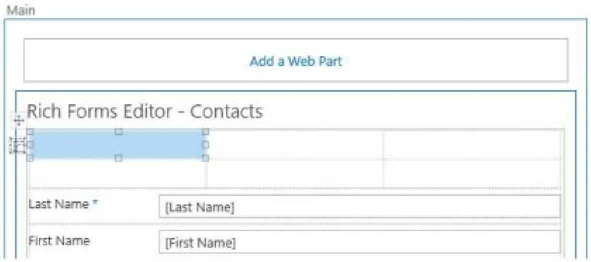
6. Table

Table is a custom action in the ribbon toolbar to add table layout into the content editor web part. Table layout allows placing fields or other elements in a familiar table grid even more easily and user-friendly. Making designing your forms as you want even faster.

Table can be inserted by selecting the needed number of rows and columns. When table is inserted, the additional ribbon tab appears with standard table options.



Table



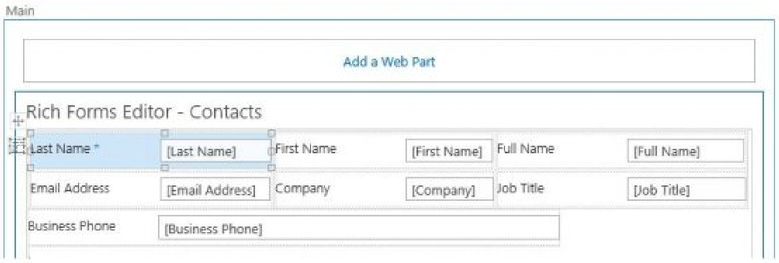
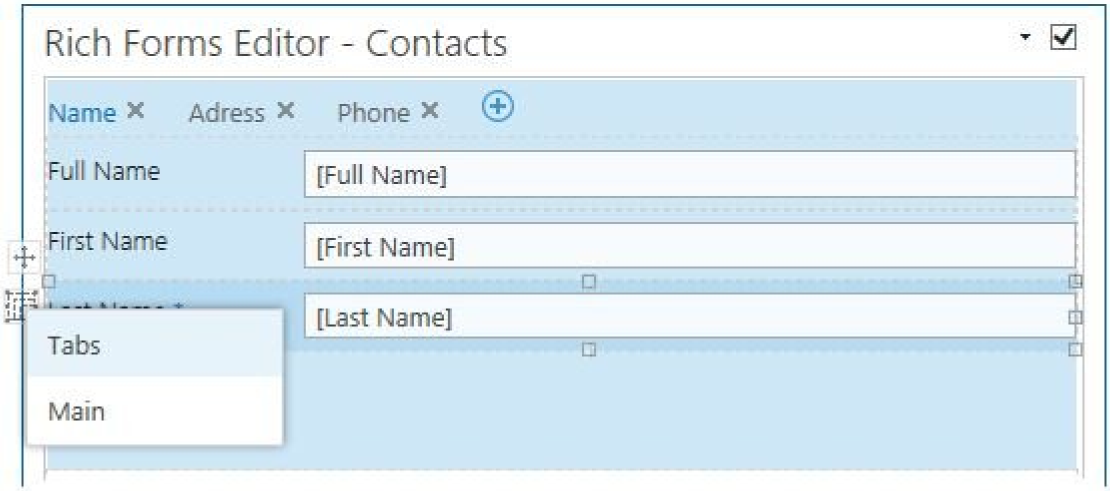


Table Layout

7. Tabs

Tabs is a custom action in the ribbon toolbar to add tabs into the content editor web part.

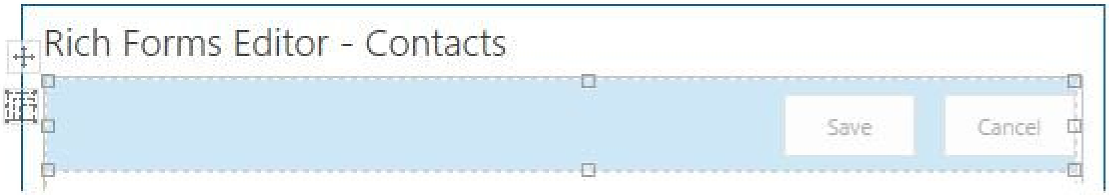
Tabs are styled by default to fit well for forms in SharePoint 2013 look, therefore they appear natural in your forms design. This group is very handy to use: tabs can be easily reordered, fields are moved to the tabs by simple drag & dropping. Each tab follows the layout rules of vertical group.



Tabs

8. Toolbar

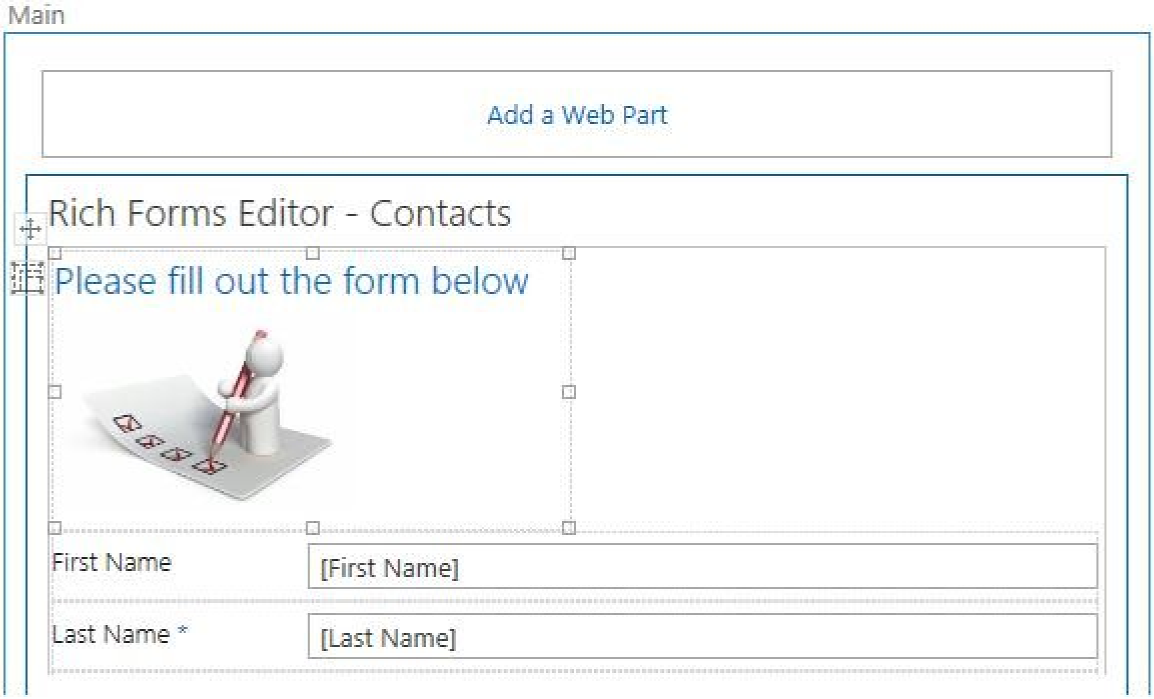
Toolbar is a custom action in the ribbon toolbar to add the form toolbar into the content editor web part. You can manage buttons separately in ungrouped toolbar.



Toolbar

9. Rich Text Box

Rich Text Box is a custom action in the ribbon toolbar to add rich text box into the form. Rich Text Box is generally used for displaying, entering and manipulating text with formatting. Now your form can be extended with the freely editable text, hyperlinks, tables or media. Fields are moved to the editable area by simple drag & dropping.



Main

Add a Web Part

Rich Forms Editor - Contacts

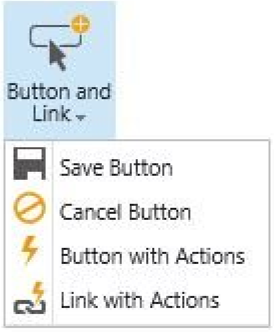
Please fill out the form below

First Name	[First Name]
Last Name *	[Last Name]

Rich Text Box

10. Button and Link

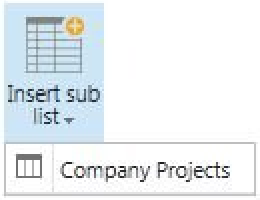
Button is a custom action in the ribbon toolbar to add the different buttons into the content editor web part. Actions executed on buttons click can be configured in Actions section of Behaviour tab.



Button and Link

11. Insert Sublist

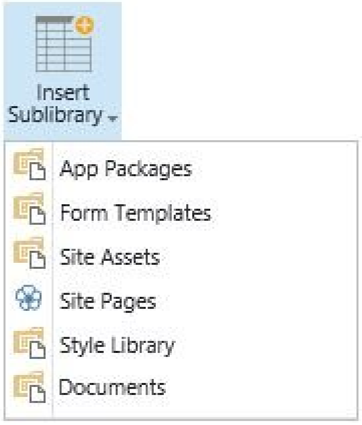
Related list view can be inserted as editable grid on new and edit form, and as non-editable grid – on display form. Available sub lists are displayed as the options of Insert sub list custom action in the ribbon toolbar. Related items are displayed based on the lookup column in the inserted list or document library.



Insert Sublist

12. Insert Sublibrary

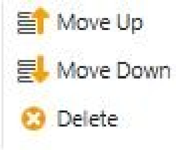
Insert Sublibrary allows providing a document library view with folders and drag and drop functionality anywhere inside your form. The sub library is editable and can be used on display and edit forms. Available sublibraries are displayed as the options of Insert Sublibrary custom action in the ribbon toolbar.



Insert Sublibrary

13. Reordering and delete actions

Lightning Forms provides an easy way of moving and deleting any UI element on the form. The following options on the ribbon allows to move a form field, as well as remove it in the design mode.

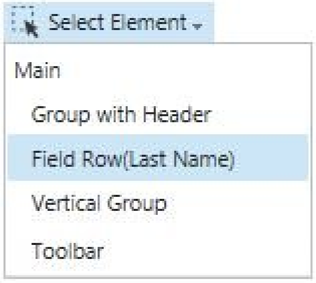


Reordering and Deleting Actions

p(banner tip).Note: In a horizontal group Move Up moves element to the left and Move Down to the right.

14. Select Element

Select Element option provides a drop-down list of all elements on the form. Selecting the option in the drop-down highlights the corresponding element on the form.

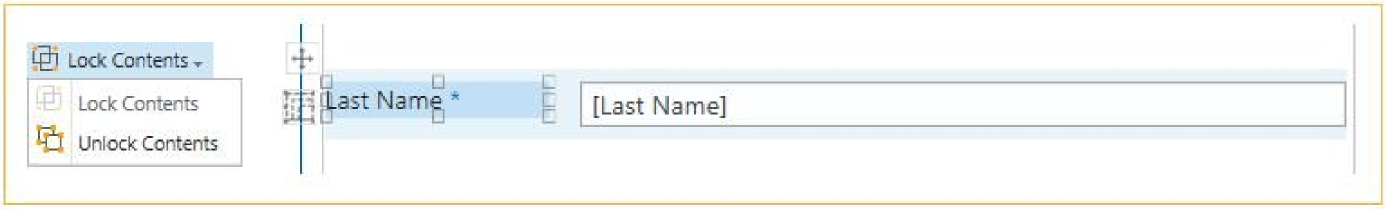


Select Element

15. Lock Contents

Rich Forms editor provides ability to lock / unlock grouped contents. This can be done by clicking one of the options under Lock Contents section. Lock action locks the selected element content: disables editing ability.

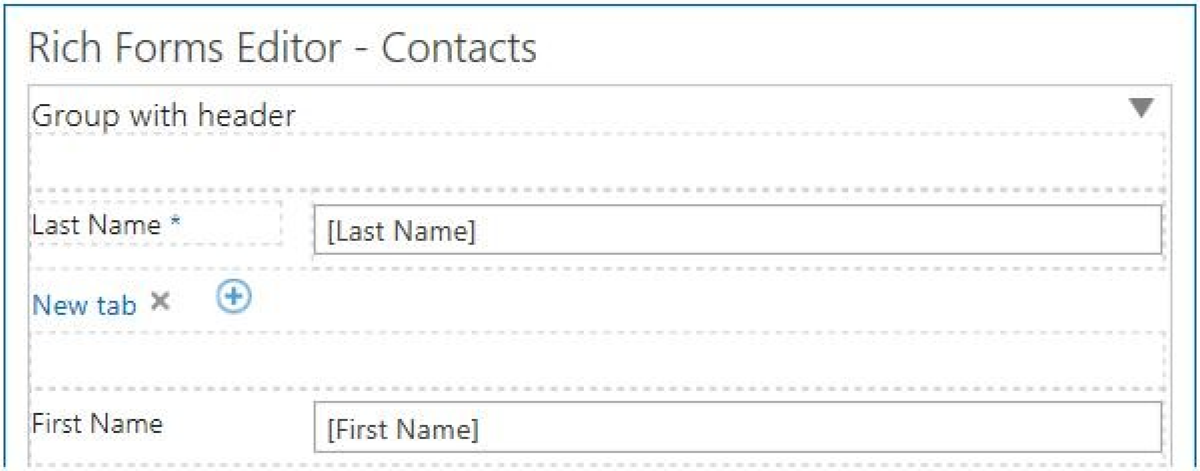
Field row is a horizontal group of field label and vertical group of field input and field description. Field row unlocking allows to manage its elements (field label, field input, field description) independently.



Lock Contents

16. Show grid lines

Show grid lines is a custom action in the ribbon toolbar to show/hide grid lines in the design display mode. It is turned on by default.



Show Grid Lines

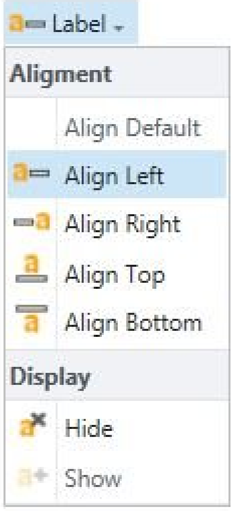
17. Label

Label is a custom action in the ribbon toolbar to align field labels into the content editor web part. In addition to simple editing of field label text and label resizing, labels also can be placed in different positions: to the left / to the right of the field, at the top / at the bottom of the field. Setting **Align Default** specifies alignment of the field label according to the context of the field row:

Horizontal / Vertical group: align setting applied to the horizontal group determines alignment of nested elements (field rows, groups, tabs).

Note: By default, horizontal group is top aligned and vertical – left aligned.

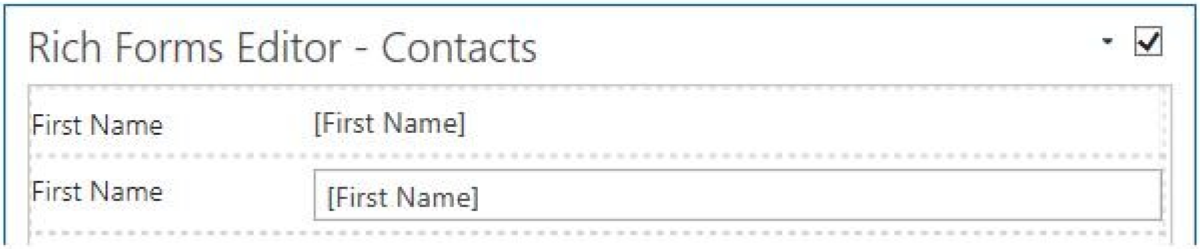
Any non-default alignment setting overrides the default one.



Label

18. Only Display

Only Display action is used to set the selected form field to only display mode, so it cannot be edited. It is useful on Edit form to have some field values in read only mode.



Rich Forms Editor - Contacts ☑

First Name [First Name]

First Name [First Name]

Only Display

19. Collapsible

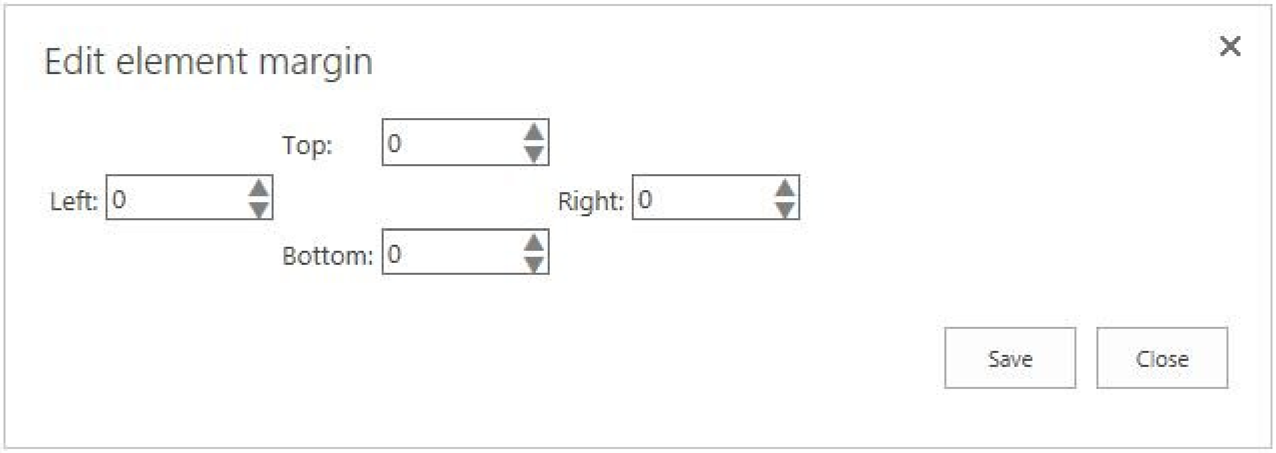
Collapsible checkbox allows to defines if Group with header will be collapsible: if an icon should appear to collapse/expand the content of this group.

20. Height, Width

Formatting actions allows to set height and width of the element, as well as of the group of elements.

21. Margin, Padding

Margin and Padding values can be set in more visual way using the Edit dialog.


Edit element margin

Top: 0

Left: 0

Right: 0

Bottom: 0

Margin, Padding

22. Export, Import

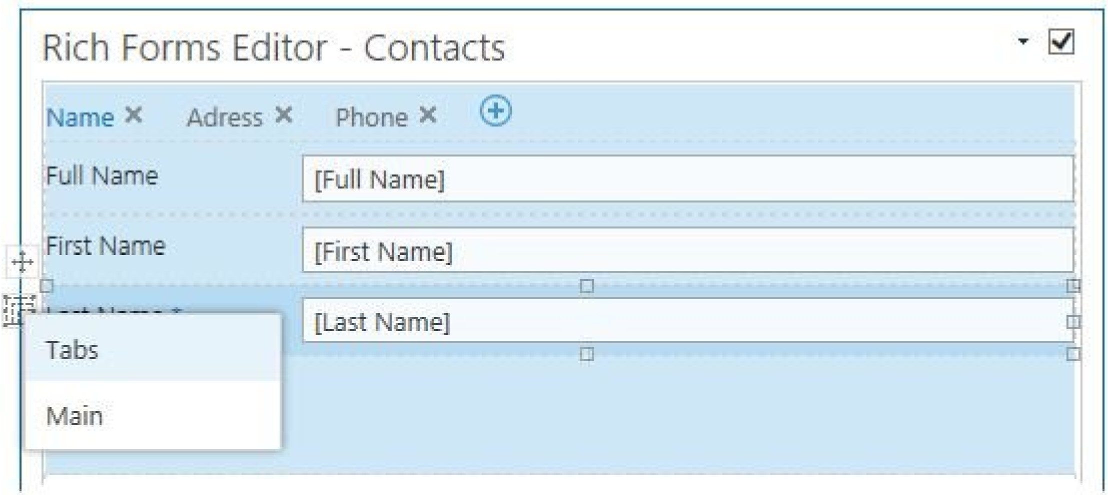
Allows exporting customized forms to a file which can be imported on the same or any other site. **Export** produces a .RFF file that contains the schema of all web parts on the form page and some additional metadata. This file can be saved and imported into another form. This is the quickest way to backup your forms and restore them on another environment.

> ✻ The list where the file is imported should have the same structure (columns) as the exported list. If there are lookups, sublists, sublibraries or other webparts on the exported form, there also should be the corresponding items on the site where the form is imported.

2.1.1. Tabs

Tabs is a custom action in the ribbon toolbar to add tabs into the content editor web part.

Tabs are styled by default to fit well for forms in SharePoint 2013 look, therefore they appear natural in your forms design. This group is very handy to use: tabs can be easily reordered, fields are moved to the tabs by simple drag & dropping. Each tab follows the layout rules of vertical group.



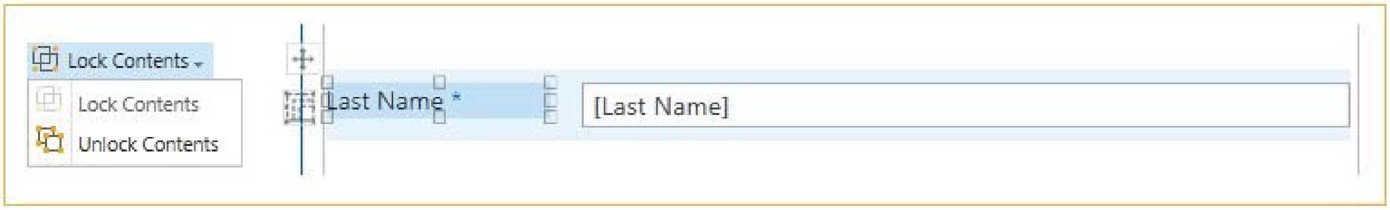
Tabs

2.1.2. Lock Contents

Lightning Forms editor provides ability to lock / unlock form contents. This can be done by clicking one of the options under Lock Contents section.

Lock action locks the selected element content: disables editing ability.

Field row is a horizontal group of field label and vertical group of field input and field description. Field row unlocking allows to manage its elements (field label, field input, field description) independently.

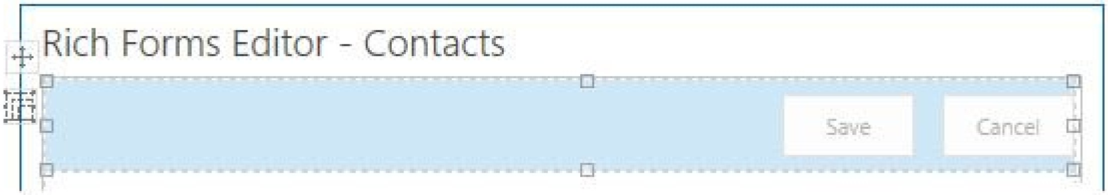


Lock Contents

2.1.3. Toolbars

Toolbar is a custom action in the ribbon toolbar to add toolbar into the content editor web part.

You can manage buttons separately in ungrouped toolbar.

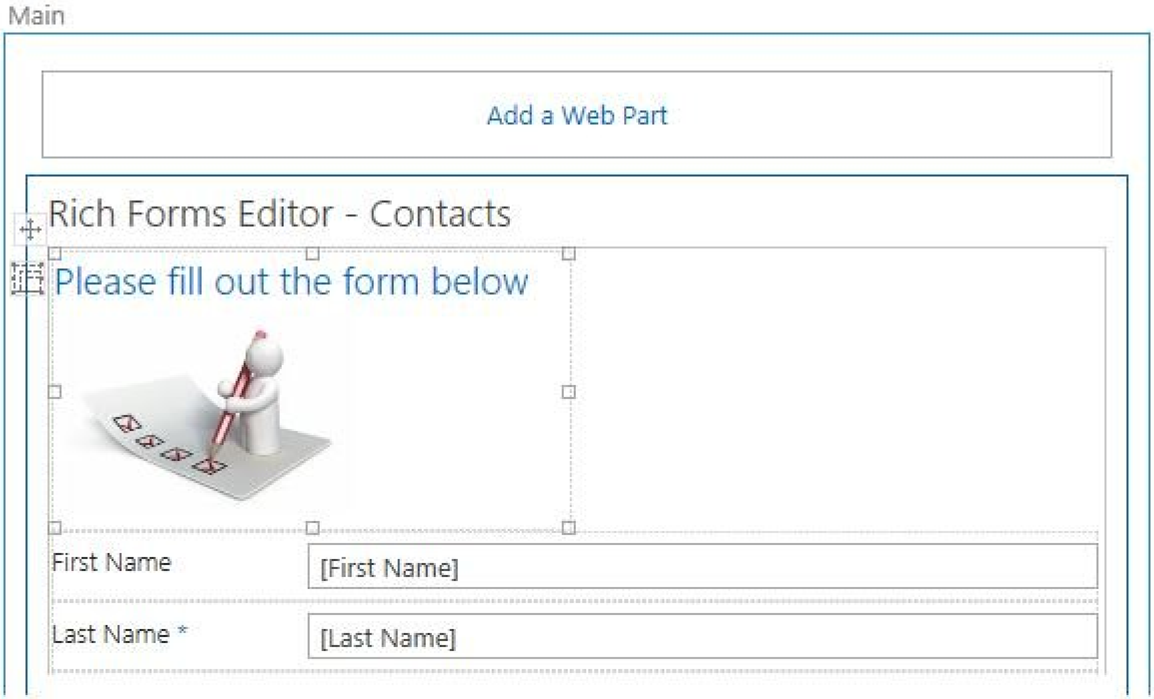


Toolbars

2.1.4. Rich Text Box

Rich Text Box is a custom action in the ribbon toolbar to add rich text box into the form.

Rich Text Box is generally used for displaying, entering and manipulating text with formatting. Now your form can be extended with the freely editable text, hyperlinks, tables or media. Fields are moved to the editable area by simple drag & dropping.



Please fill out the form below

First Name [First Name]

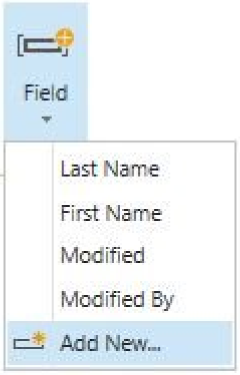
Last Name * [Last Name]

Rich Text Box

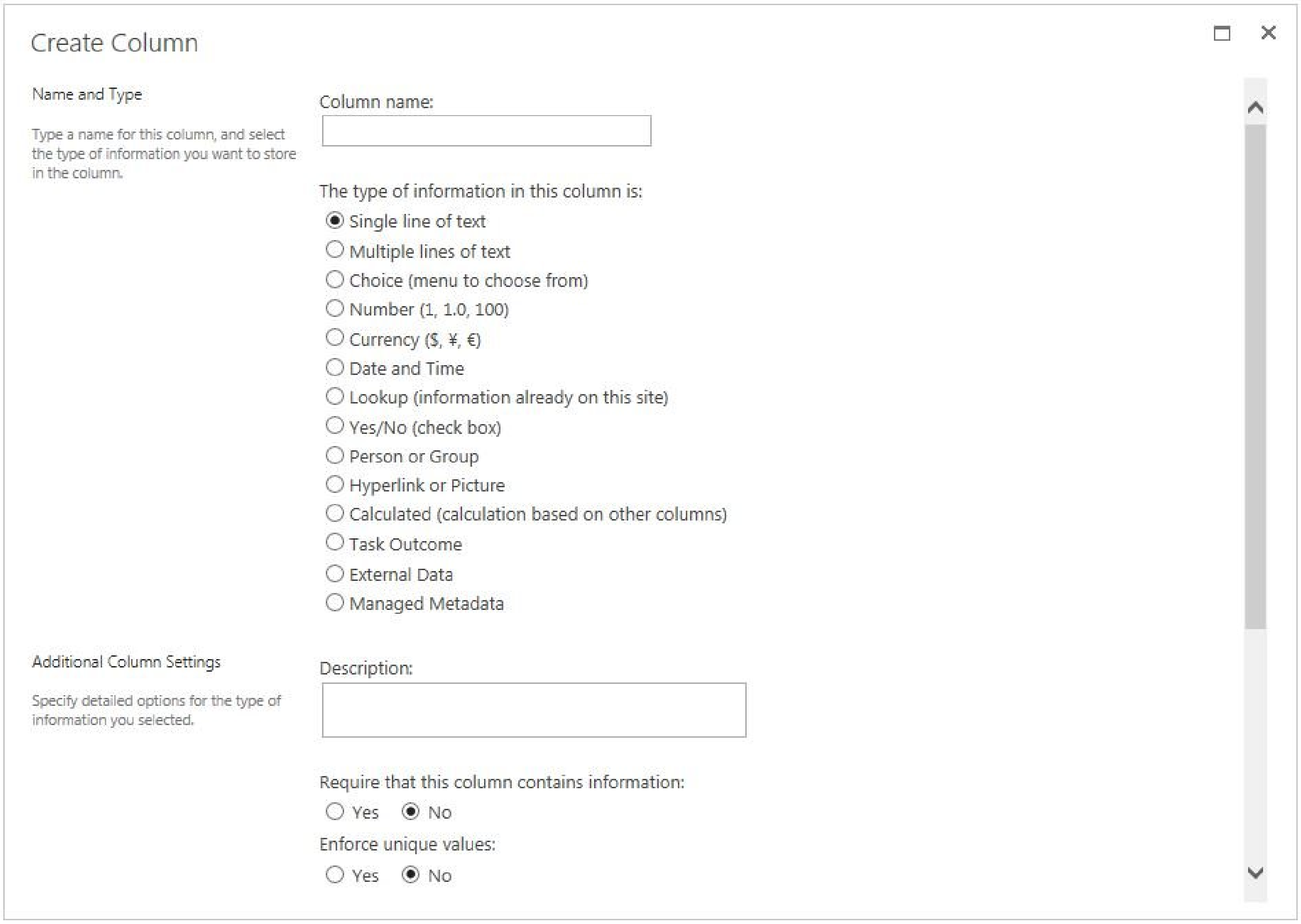
2.1.5. Insert Fields

Field is a custom action in the ribbon toolbar to add fields into the content editor web part. It allows you to add any available for the current form field.

It also provides an ability to add new field directly form the current page: clicking **Add New** button initiates opening of **Create Column** dialog. New field is immediately added to the form.



Insert Field



Create Column

Name and Type

Type a name for this column, and select the type of information you want to store in the column.

Column name:

The type of information in this column is:

● Single line of text
○ Multiple lines of text
○ Choice (menu to choose from)
○ Number (1, 1.0, 100)
○ Currency ($, ¥, €)
○ Date and Time
○ Lookup (information already on this site)
○ Yes/No (check box)
○ Person or Group
○ Hyperlink or Picture
○ Calculated (calculation based on other columns)
○ Task Outcome
○ External Data
○ Managed Metadata

Additional Column Settings

Specify detailed options for the type of information you selected.

Description:

Require that this column contains information:
○ Yes ● No

Enforce unique values:
○ Yes ● No

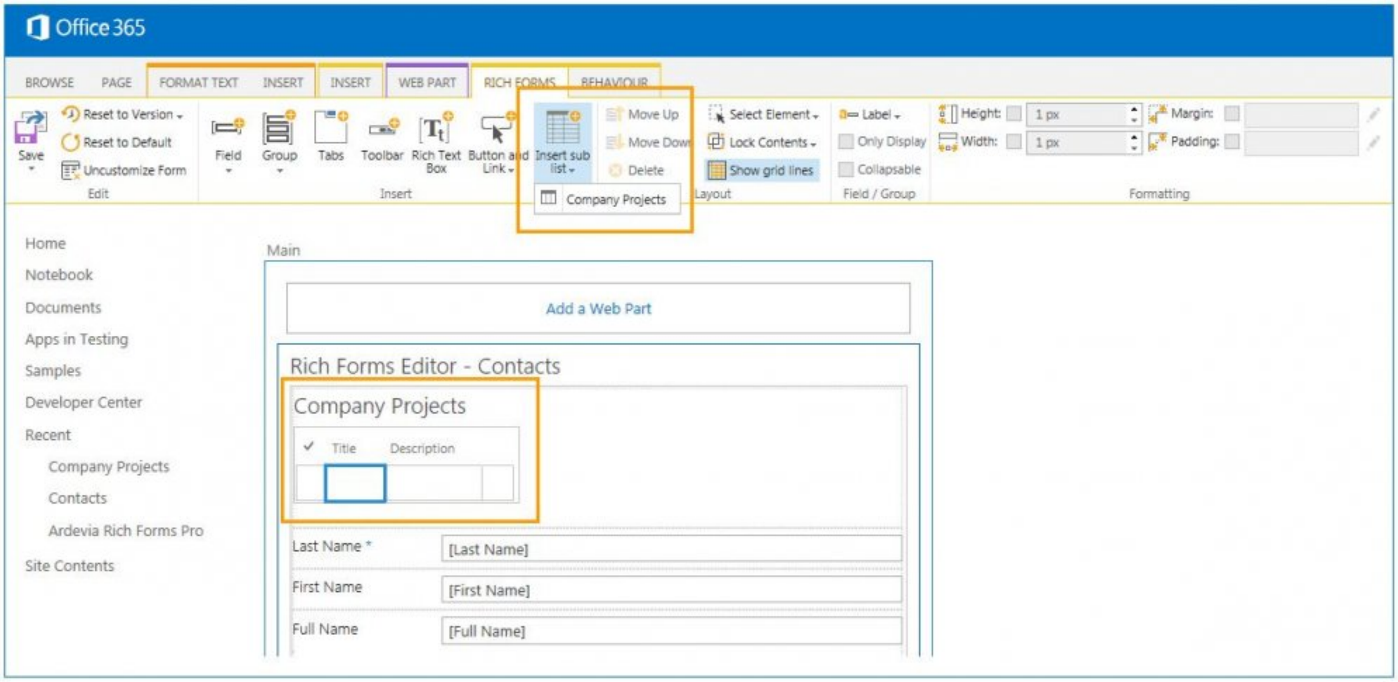
Field Type

2.1.6. Insert Sub List

Using a sublist allows providing a grid view of a related list anywhere inside your form. Related lists are lists that contain a lookup column to the list or library the current form is for. The sublist is editable and can be used on new, edit and display forms.

Available sublists are displayed as the options of Insert sublist custom action in the ribbon toolbar. Only lists or libraries are displayed with a lookup to the form’s list.

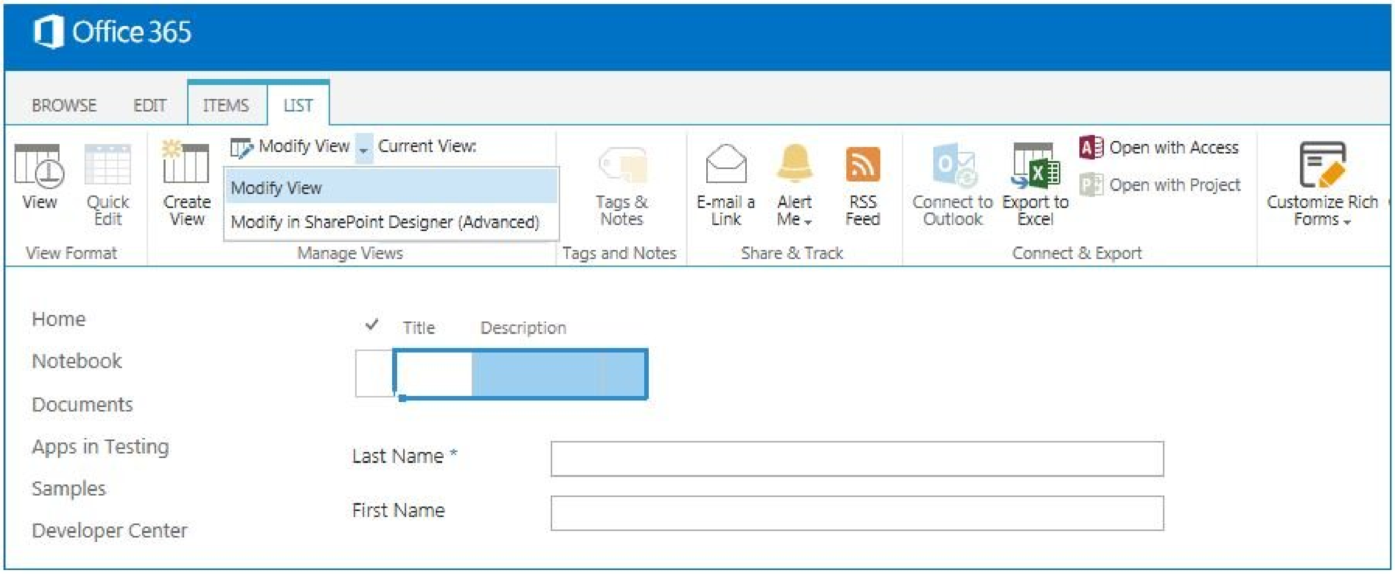
> ✻ The lookup to the form’s list is not shown in the grid.



Insert Sublist

The view of the inserted sublist can be edited in the same way as an out-of-the-box list view web part (to change things like query, fields, etc.):

- Select the row on the list view
- Click Modify View on the List tab of the ribbon.

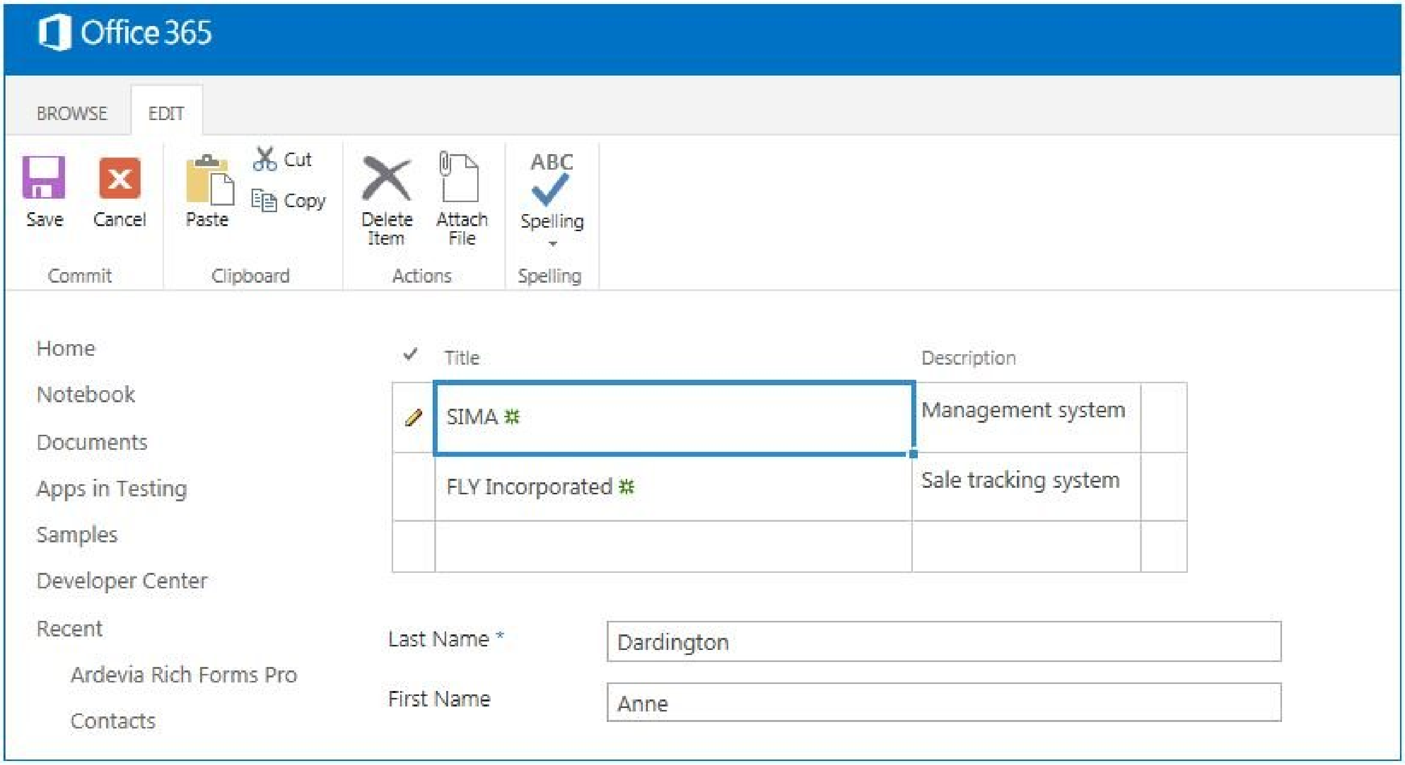


✓	Title	Description

Last Name *	
First Name	

Modify View

✻ If tabular view is turned off on a list view used in a sub list, editing items in sub lists is disabled.



✓	Title	Description
✎	SIMA ✻	Management system
	FLY Incorporated ✻	Sale tracking system

Last Name *	Dardington
First Name	Anne

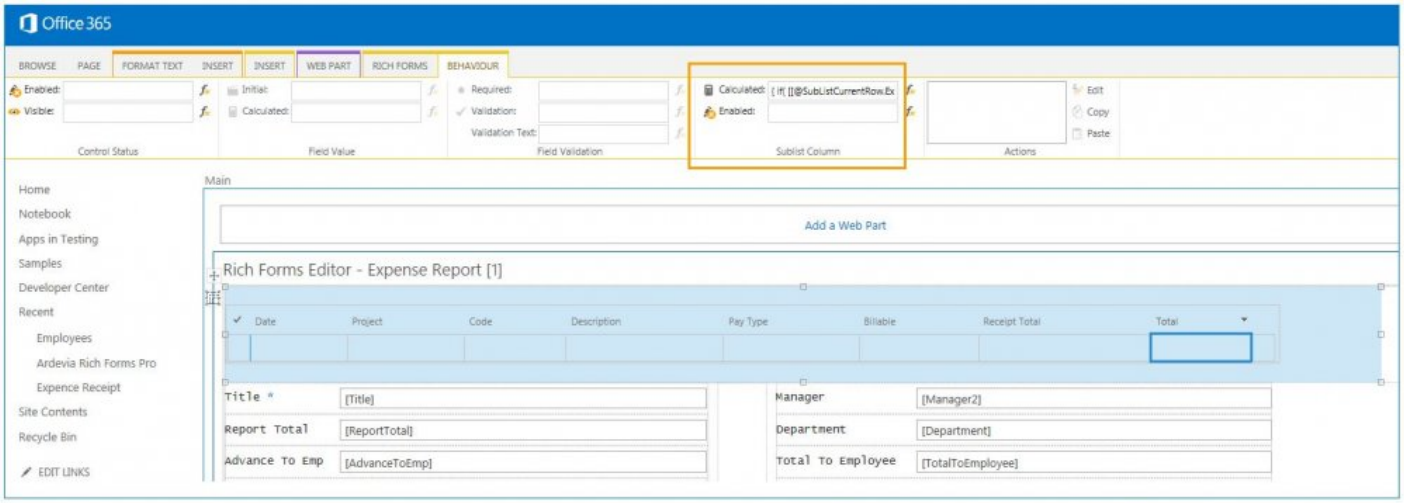
Grid View

On new and edit forms the data edited in the sublist is only stored when the whole form is saved.

2.1.6.1. Sublist Column Calculation

Lightning Forms supports calculated and enabled expressions for column cells in a sublist grid. They can be configured on Behaviour tab in Sublist Column section when a column is selected in the form editor and can be used on all available form types.

> ✻ Unlike the value and validation expressions for form fields, calculated and enabled expressions for sublist columns are available on Display forms too, since the grid is editable.



Calculated: { If([[@SubListCurrentRow.Ex

Title *	[Title]	Manager	[Manager2]
Report Total	[ReportTotal]	Department	[Department]
Advance To Emp	[AdvanceToEmp]	Total To Employee	[TotalToEmployee]

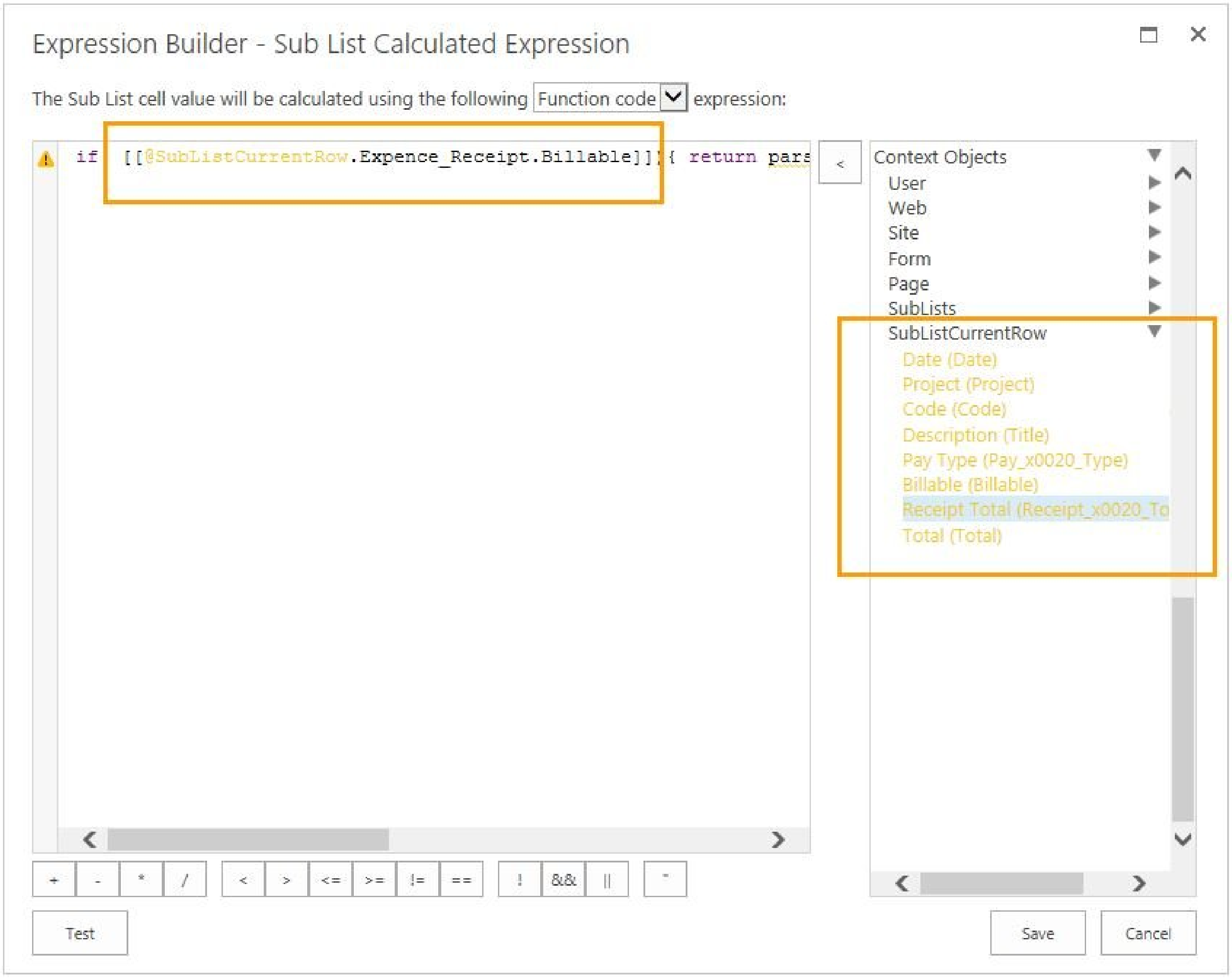
Sublist Column Calculation

1. Calculated

Calculated expression provides a flexible way to dynamically recalculate the value in the column while the user is entering data in the form or other columns of the sublist. Whenever the value of a field referenced in the calculated expression is changed the expression is recalculated and the dependent field’s value updated with the result.

The calculation can be based on form field values, as well as on values of other cells in the current row. To reference other cells in the current row SubListCurrentRow placeholders are available under the Context Objects.

✻ It is possible to set initial value for sublist column using calculated expression relied on main list field (New form).



Expression Builder - Sub List Calculated Expression

The Sub List cell value will be calculated using the following Function code expression:

```
if [[@SubListCurrentRow.Expence_Receipt.Billable]]{ return pars
```

Sublist Calculated Expression

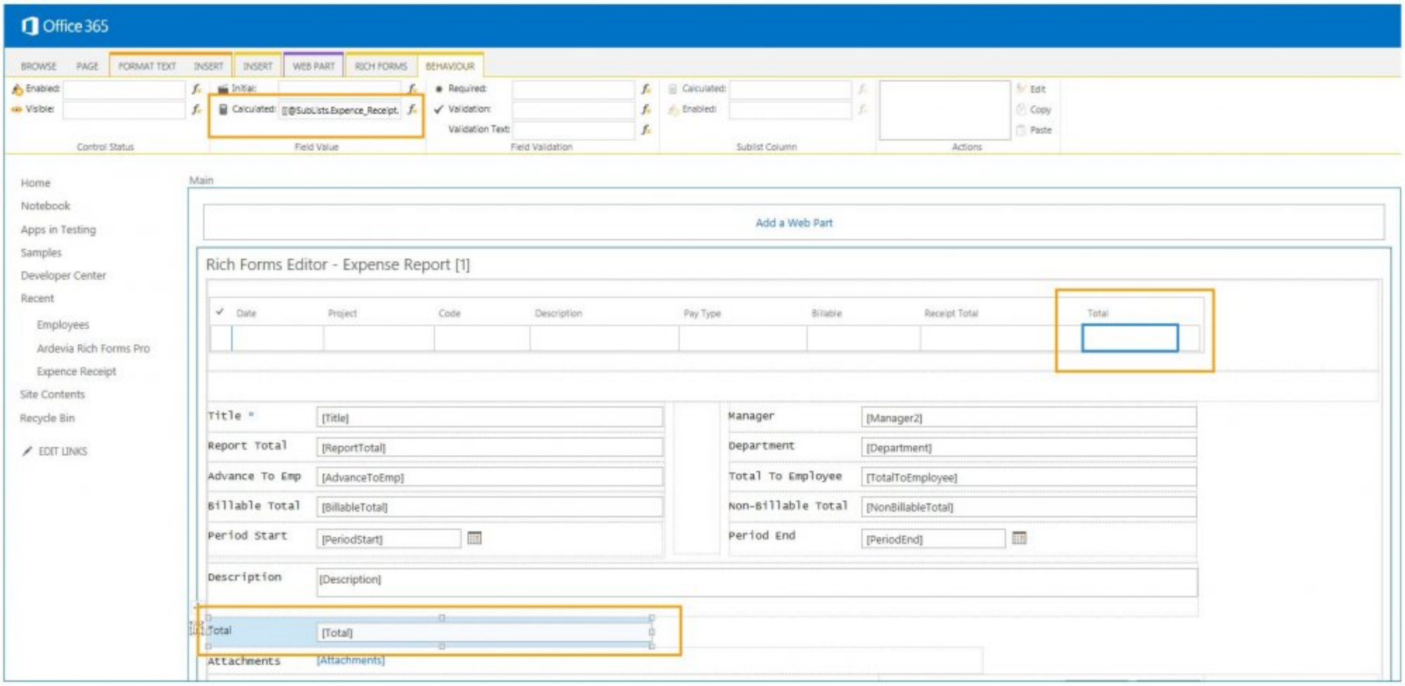
2. Enabled

Enabled expression allows to enable or disable any cell in the sublist at run-time based on some condition. For example, you can disable cells that are not allowed to be edited depending on other cell values in the current row or form values.

For more information about expressions used for configuring behaviours, see Expressions topic.

2.1.6.2. Aggregation over Sublists

There is often a need to perform some operations over all values of a sublist column. Using aggregation functions in expressions sums, counts, mins, maxs, averages or custom logic can be calculated over all values in a column of a sublist. These expressions can be used as field values or to set fields visible or enabled depending on values in the sublists.

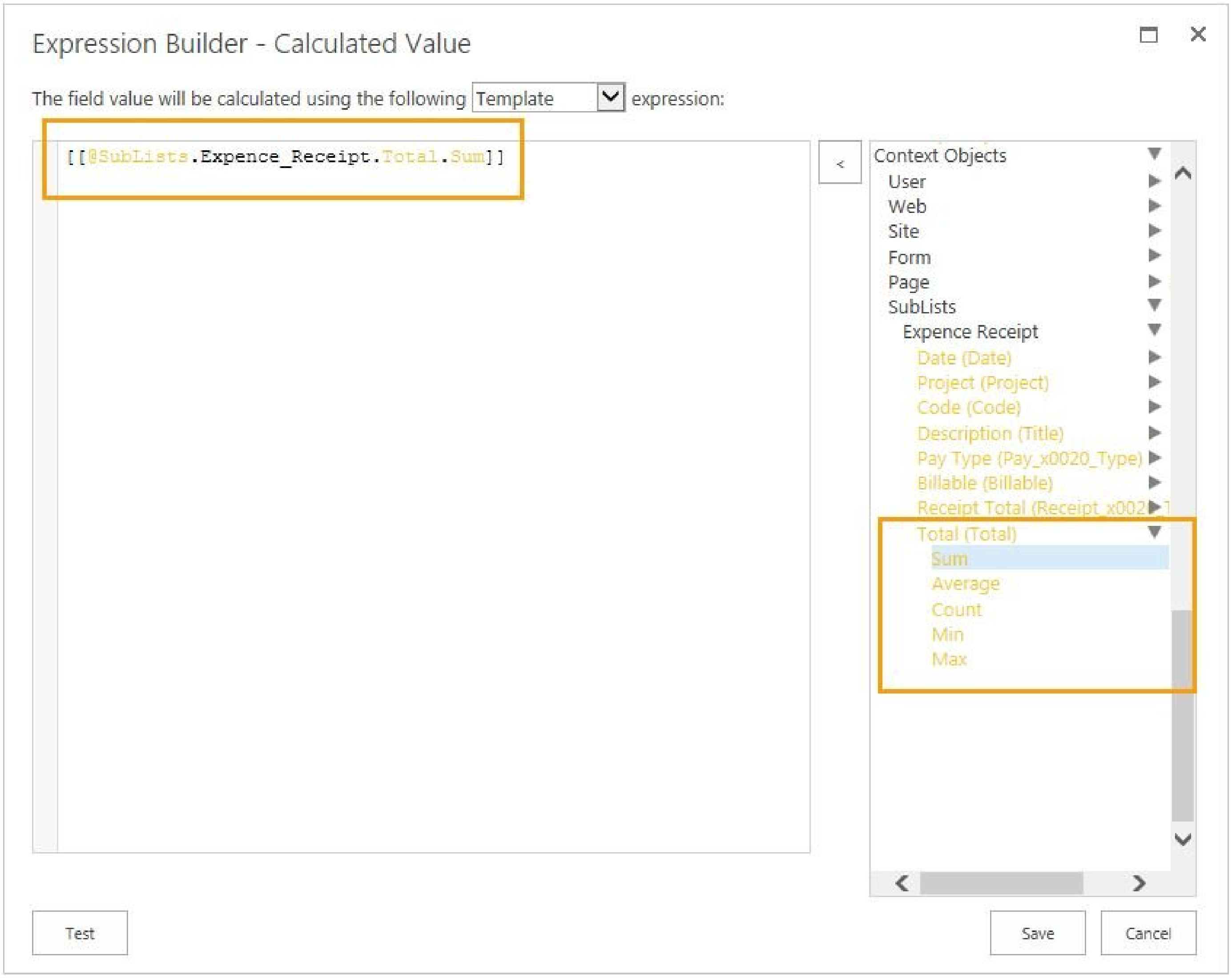


Calculated: [[@SubLists.Expence_Receipt.

Rich Forms Editor - Expense Report [1]

Title *	[Title]	Manager	[Manager2]
Report Total	[ReportTotal]	Department	[Department]
Advance To Emp	[AdvanceToEmp]	Total To Employee	[TotalToEmployee]
Billable Total	[BillableTotal]	Non-Billable Total	[NonBillableTotal]
Period Start	[PeriodStart]	Period End	[PeriodEnd]
Description	[Description]		
Total	[Total]		
Attachments	[Attachments]		

Aggregation over Sublists



Expression Builder - Calculated Value

The field value will be calculated using the following Template expression:

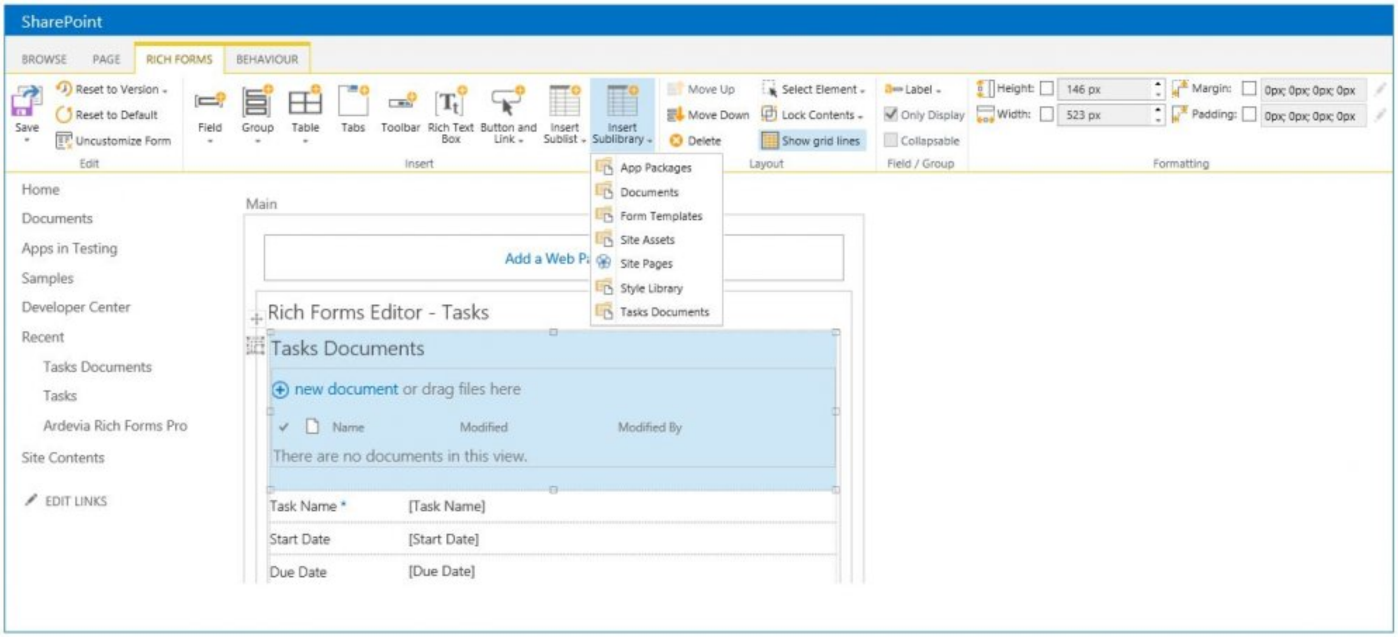
```
[[@SubLists.Expence_Receipt.Total.Sum]]
```

Expression Builder Sublist

2.1.7. Insert Sublibrary

Using a sublibrary allows providing a document library view with folders and drag and drop functionality anywhere inside your form. Rich Forms will automatically organize all documents in a folder per parent item generated on the document library.

All available libraries are displayed to be inserted using the Insert Sublibrary custom action of the ribbon toolbar.



Height: 146 px
Width: 523 px
Margin: 0px; 0px; 0px; 0px
Padding: 0px; 0px; 0px; 0px

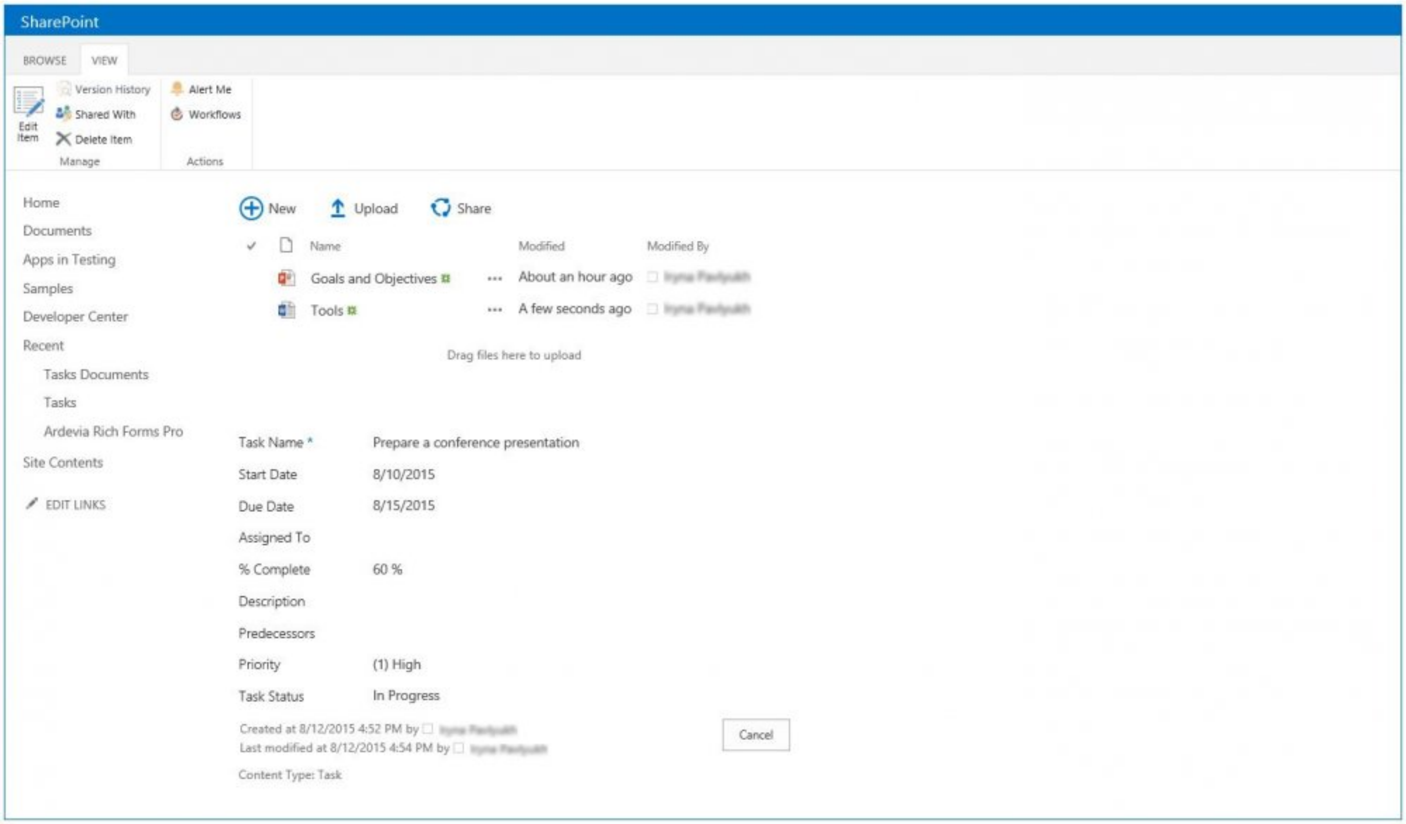
Task Name * [Task Name]
Start Date [Start Date]
Due Date [Due Date]

Insert Sublibrary

The view of the inserted sublibrary can be edited in the same way as an out-of-the-box list view web part (to change things like query, fields, etc.):

- Select some document
- Click Modify View on the List tab of the ribbon.

The view behaves at run-time just like the out-of-the-box view, including filtering, sorting and adding columns functionalities.
The items are saved without saving the form.



Name	Modified	Modified By
Goals and Objectives	About an hour ago	[redacted]
Tools	A few seconds ago	[redacted]

Task Name *	Prepare a conference presentation
Start Date	8/10/2015
Due Date	8/15/2015
Assigned To	
% Complete	60 %
Description	
Predecessors	
Priority	(1) High
Task Status	In Progress

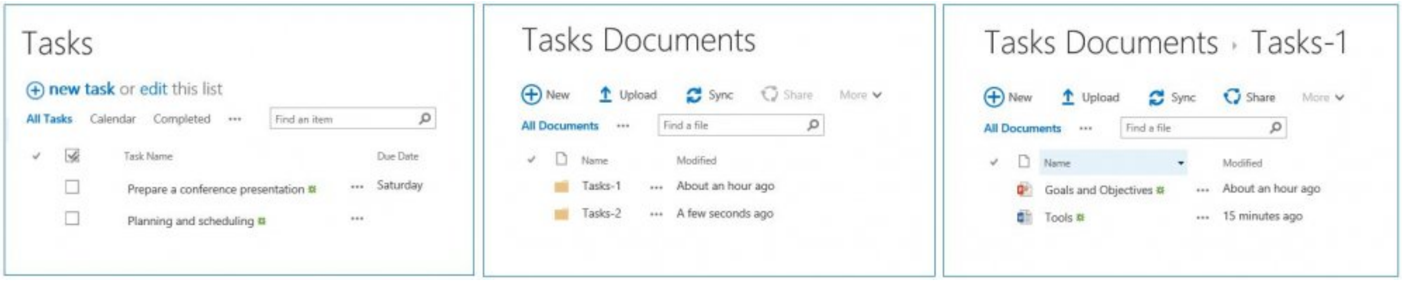
Created at 8/12/2015 4:52 PM by [redacted]
Last modified at 8/12/2015 4:54 PM by [redacted]

Content Type: Task

Sublibrary saving

Folder structure

Folders per list item in the sublibraries are created automatically when the corresponding item's form is opened. All related files are saved inside this folder.
The folders are named using the list's name and item's ID.



Folder Structure

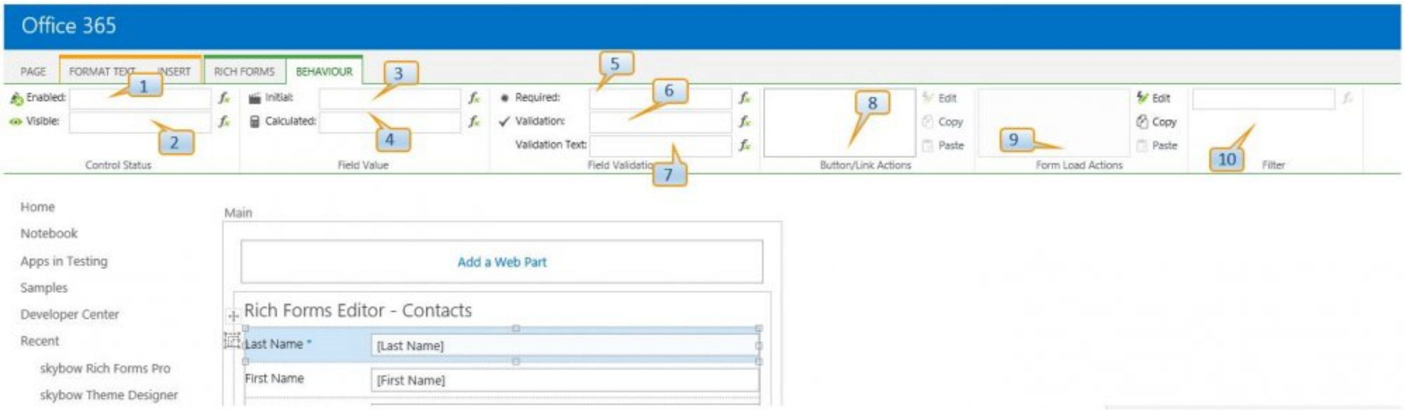
2.2. Behavior Tab

Behaviour Tab Features

Behaviour tab contains controls for configuring complex business logic. Once you open your form in design mode, it will appear after standard SharePoint tabs and the Rich Forms tab.

Flexibly configurable, expression-based behaviours provide an easy way to make a field or form section dynamically show or hide or become enabled or disabled depending on other inputs, set default values and even recalculate them while the user is entering data in the form.

For more information about expressions used for configuring behaviours, see Expressions topic.



Behaviour Tab

1. Enabled

Enabled expression allows to enable or disable any selectable controls at run-time based on some condition.

For example, you can disable controls that are not allowed to be edited depending on other field values or disable a button to prevent the user from clicking it depending on his group memberships.

This option is available on all form types: New, Edit, Display.

2. Visible

Visible expression allows to change the visibility of any selectable controls at run-time based on some condition.

For example, you can hide or show form controls based on conditions. As an example, a field can be shown only when the field ‘Status’ is set to ‘Active’: =[[Status]]==‘Active’

This option is available on all form types: New, Edit, Display.

3. Initial

Initial value expression allows to assign a default value to a field. This can be a constant or a calculated expression.

For example, you can set current date as a default value: ‘= new Date()’

This option is available only on New forms.

4. Calculated

Calculated expression provides a flexible way to dynamically recalculate the value while the user is entering data in the form. Whenever the value of a form field referenced in the calculated expression is changed the expression is recalculated and the dependent field’s value set with the result.

For example, you can use it to perform a mathematical calculation – multiplying the values contained in two fields in order to produce a Total Amount in the current field: ‘=[[Price]]*[[Count]]’

This option is available only on New and Edit forms.

5. Required

Required expression is used to specify the condition when a field’s value must be set before the form can be submitted.

For example, you can specify that users must fill in an “Explanation” field when the leave requests “Reason” field is “Other”: = [[Reason]] == “Other”

This option is available only on New and Edit forms.

6. Validation

Validation expression provides an easy way to define a validation rule. Using expressions even complex rules can be defined to specify required form input. These can depend on the entered data. When the validation expression evaluates to false the form can not be saved and the message evaluated using Validation Text (see below) is shown.

For example, when the “Reason” field is “Compensation” the “Duration” field can not be higher than 3 days:
= ([[Reason]] != “Compensation”) || ([[Duration]] <= 3)

This option is available only on New and Edit forms.

7. Validation Text

Validation Text expression is used to specify the text that is shown when the Validation expression (see above) fails (t.i. returns false).

8. Button/Link Actions

Actions can be defined on buttons or links to specify what should happen when these are clicked.

The Actions Builder allows you to edit which actions in which order and under which conditions should be executed.

The following action types are available:

Actions Builder
Add List Item
Delete List Item
Execute Script
HTTP Request
Open List Form
Open Web Page
Redirect to URL
Reload Page
Save Form
Send Mail
Show Message

Start Workflow
Update List Item

Most actions require some parameters which can be configured in the Actions Builder. In most cases this configuration can depend on form data or contextual information using expressions. Thus even complicated data-dependent logic or navigation can easily be implemented without any programming knowledge.

9. Form Load Actions

It is also possible to define actions that are executed when the form is loading. All previously described action types are available.

10. Lookup Filtering

Filter criteria can be set when some lookup field is selected.

CAML Query Editor is accessible when Edit Query button is clicked.

11. Sublist Column Expressions

Rich Forms supports calculated and enabled expressions for column cells in a sublist grid. They can be configured when a column is selected in the form editor and can be used on all available form types.

For more information see here.

2.2.1. Expressions

Introduction

Business and UI-logic on forms need to be flexible to fulfil all the possible needs and wishes. Rich Forms provides a powerfull expression language to enable configuring forms freely. Throughout the forms editor expressions can be used to dynamically calculate a value or text given a specific context.

To cater for different uses any expression can be defined in any of the following forms: Template Expressions, Value Expressions, Function Expressions.

Template expressions

Text templates are a special form of expressions in which placeholders are replaced by string representation (always returns a string, placeholders are always converted to string before inserted). Placeholders can contain field paths [[CaseFile.Title]] or expressions [[= Context.Web.Title]] (begin with equals sign).

This form allows to define textual values in a simpler, more abreviated and suitable manner than the other forms. As an example, we can write ‘[[Firstname]] [[Lastname]]’ for a calculation of a text field value, instead of needing the more complex value expression (‘=[[Firstname]] + “ “ + [[Lastname]]’).

Syntax: Use plain text containing placeholders to insert dynamically calculated text.
Example: Use the expression ‘Welcome [[Firstname]] [[Lastname]]!’ to calculate the text ‘Welcome Mary Hunter!’ when the values of field Firstname and LastName are ‘Mary’ and ‘Hunter’

Assignment expressions

With value expressions we can define texts and values to be calculated using JavaScript-based assignment terms.

Syntax: The “=” sign at the beginning of expressions signals that it is an assignment expression, otherwise it is evaluated as a template expression. When the assignment expression type has been selected in the expression builder the preceding equals does not need to be typed though. Use placeholders in double square brackets to access a field value or context object that is dynamically evaluated.
Example: = [[MyTest]] + ‘ is calculated value’

Function Expressions

Using function expressions more complex calculations can be defined. The expression allows using the full scope of JavaScript code as in bodies of JavaScript functions. Values must be returned using ‘return ;’ statements.

Syntax: Starting with “{” signals that it is a function body expression. In the expression builder the curly brackets don’t need to be typed though. Except for the use of placeholders in double square brackets to inject field values and context information the syntax is the same as used in JavaScript function bodies. Example: { var myValue = [[Title]]; if (myValue.indexOf(‘Test’) > -1) return “Is Test”; else return myValue; }

2.2.1.1. Expression Examples

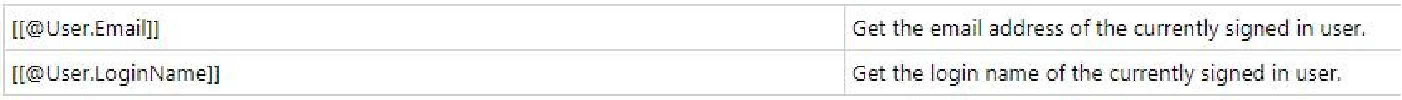
Initially setting date time fields

= new Date()	Set the current date time (can also be done just by using SharePoint default column value available in column’s configuration)
= new Date(2014, 5, 3)	Set the date to 3rd June 2014. Note: The month (2nd) parameter is 0-based, so January is 0 and December is 11.
= new Date().addDays(10)	Set the date to current date (and time) plus 10 days. Use addDays(-n) to subtract n days.
= new Date().addHours(4)	Set the date to 2 hours in future.
= new Date().addYears(-1)	Set the date a year ago from today.

Calculating date time fields

=[[SomeOtherDate]]	Get the date time field of another date time field. Note: Use value expression when setting fields using calculated expressions.
=[[SomeOtherDate]].addDays(10)	Get the date time field of the SomeOtherDate column plus 10 days.
{ if ([[ContractType]] "Monthly") return [[StartDate]].addMonths(1); else if ([[ContractType]] “Quarterly”) return [[StartDate]].addMonths(3); else if ([[ContractType]] "Semi-Annually") return [[StartDate]].addMonths(6); else if ([[ContractType]] “Annually”) return [[StartDate]].addYears(1); else return null; }	Get a date depending on value of field ContractType and StartDate. When ContractType is “Monthly” the date is calculated as StartDate plus 1 month, when ContractType is “Quarterly” the date is calculated as StartDate plus 3 months, etc. If ContractType does not match any of the checked values no date is returned.As an example this function expression could be used as a calculated expression to set the RenewalDate field of a contract item.

Retrieving information for current user



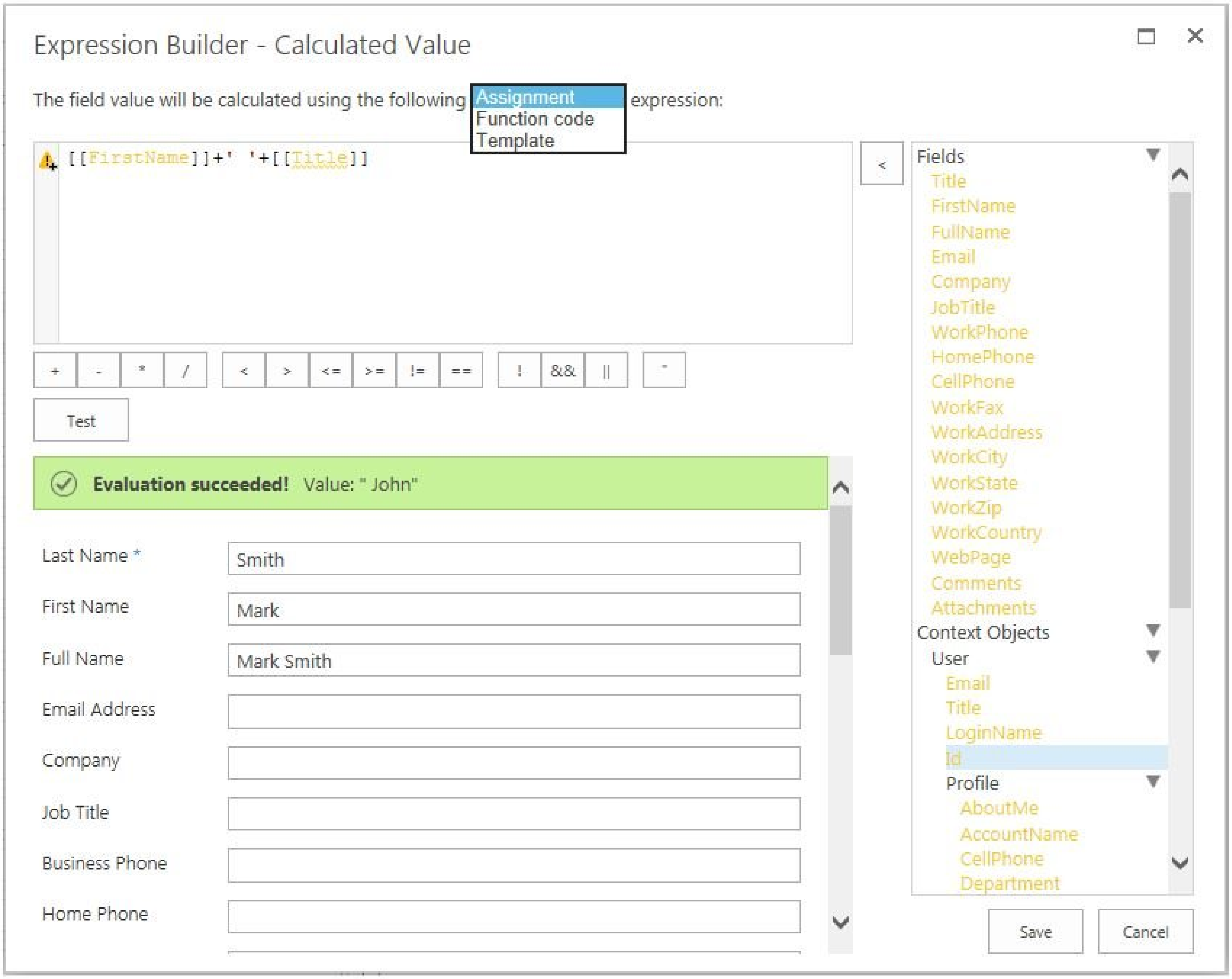
[[@User.Email]]	Get the email address of the currently signed in user.
[[@User.LoginName]]	Get the login name of the currently signed in user.

Retrieving information for the user selected for a field

[[SomeUserField.Email]]	Gets the email address of the user selected for the field SomeUserField.
[[SomeUserField.LoginName]]	Gets the login name of the user selected for the field SomeUserField.

2.2.1.2. Expression Builder

Use the Expression Builder to easily write and verify expressions.



Expression Builder - Calculated Value

The field value will be calculated using the following Assignment expression:

```
[[FirstName]]+' '+[[Title]]
```

Evaluation succeeded! Value: " John"

Last Name *	Smith
First Name	Mark
Full Name	Mark Smith
Email Address	
Company	
Job Title	
Business Phone	
Home Phone	

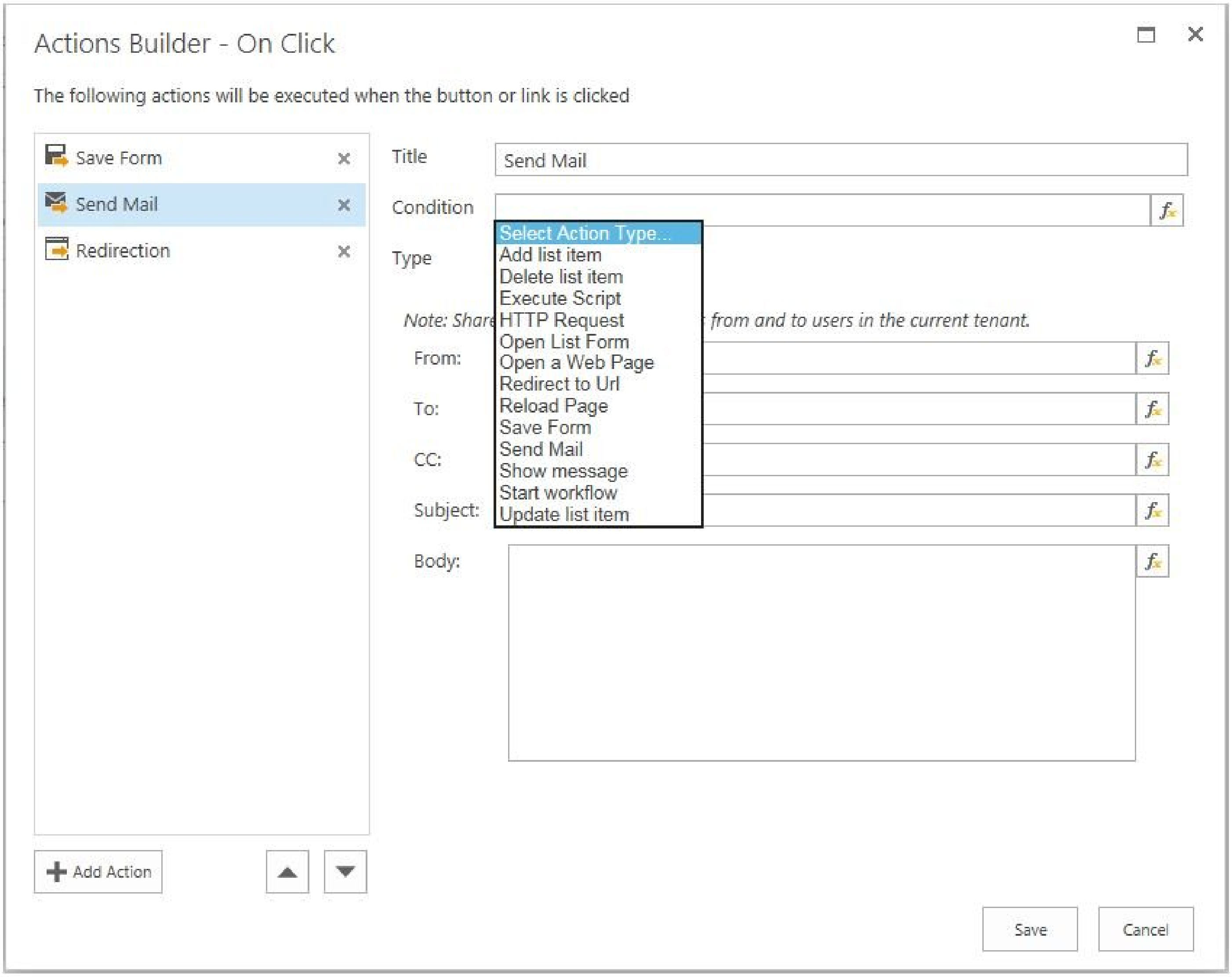
Expression Builder

Use the dropdown to select between the different expression types. Enter the expression in the text editing box.

To assist inserting placeholders all available fields and context object placeholders on the right hand panel are shown and can be added using the arrow button or double click. Afterwards expression can be checked by clicking the Test button. The result is then shown along with the editable form. The forms buttons are disabled in this mode, but using the Test button again expression can be re-evaluated.

2.3. Actions Builder

Actions Builder is a tool that allows you to specify the actions for buttons and links controls.
The dialog consists of 2 parts: list of actions and list of properties according to selected action.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: Send Mail

Note: Share from and to users in the current tenant.

Actions Builder

To add an action, click the ‘Add Action’ button.

All actions have the following general properties:

Title: Title to use for the action in the list of actions.
Type: Choose the action type to use when executing the action.
Depending on types, each action link has additional properties.

Actions are executed one by one as they are shown in the left hand list. To change order of the actions, use appropriate Up and Down arrow buttons.

The following action types are available:

Actions Builder
Add List Item
Delete List Item
Execute Script
HTTP Request
Open List Form
Open Web Page
Redirect to URL
Reload Page
Save Form
Send Mail
Show Message
Start Workflow
Update List Item

2.3.1. Add List Item

Add List Item action allows adding a new item dynamically to the specified list.

When action link type “Add List Item” is selected, the following additional properties can be specified:

- List: a list where a new item should be added. It is possible to select the list from the current or any other site, by selecting the list in ‘Select List or Library’ dialog or typing the server relative URL to a list.

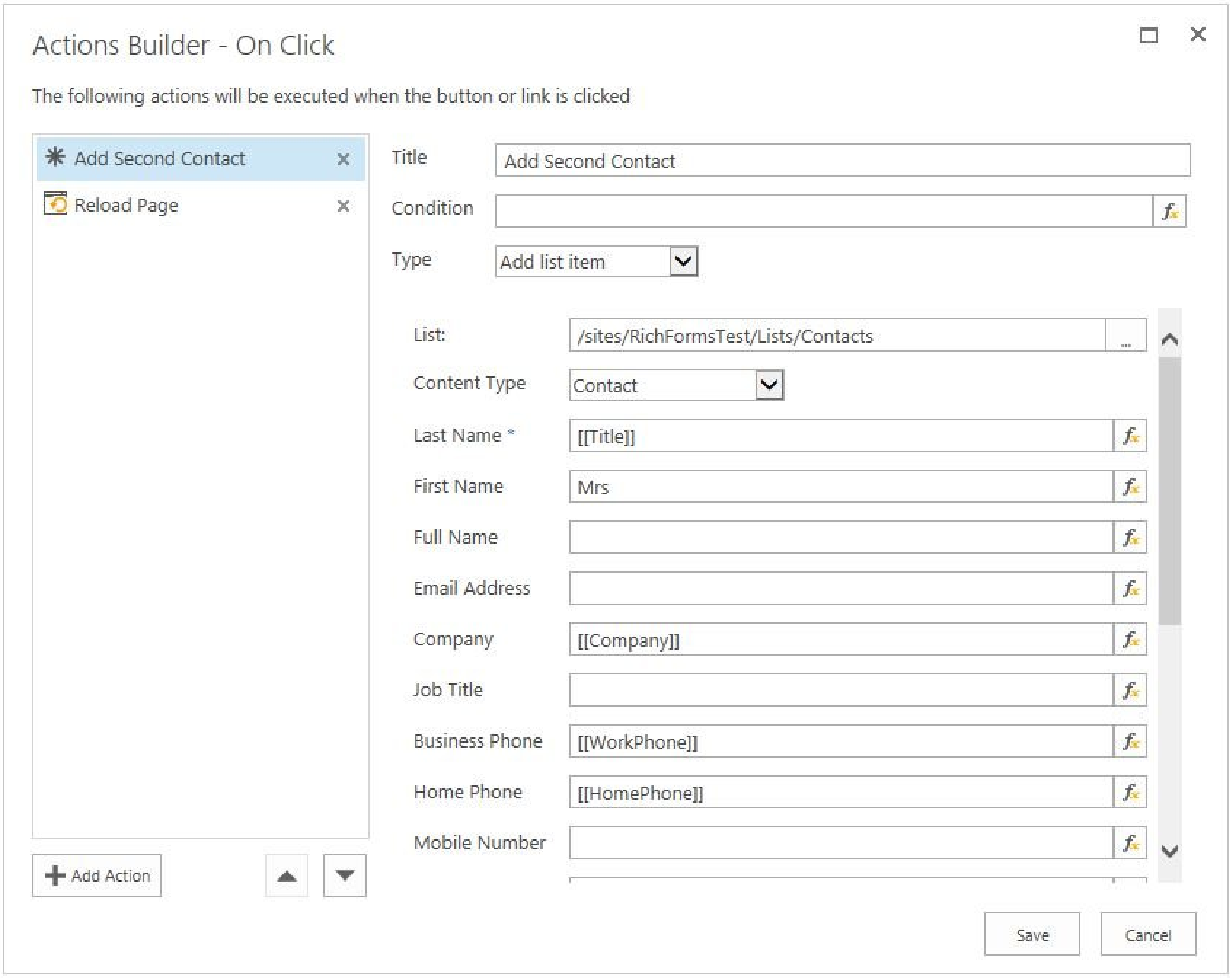
✻ You cannot add an item (document) to a library using this action type.

- Content Type: Allows choosing the content type of the item to be added.
- Fields specific for Content Type: Depending on the content type, all relevant fields are displayed at the bottom, so that for each field the value to be set can be configured.

When the Add List Item action is executed, the new item is added immediately setting the configured field values.

✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder.

✻ Only specified fields will be updated. If you leave the field blank, it will not be changed.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

- Add Second Contact ×
- Reload Page ×

Title	Add Second Contact
Condition	
Type	Add list item

List:	/sites/RichFormsTest/Lists/Contacts
Content Type	Contact
Last Name *	[[Title]]
First Name	Mrs
Full Name	
Email Address	
Company	[[Company]]
Job Title	
Business Phone	[[WorkPhone]]
Home Phone	[[HomePhone]]
Mobile Number	

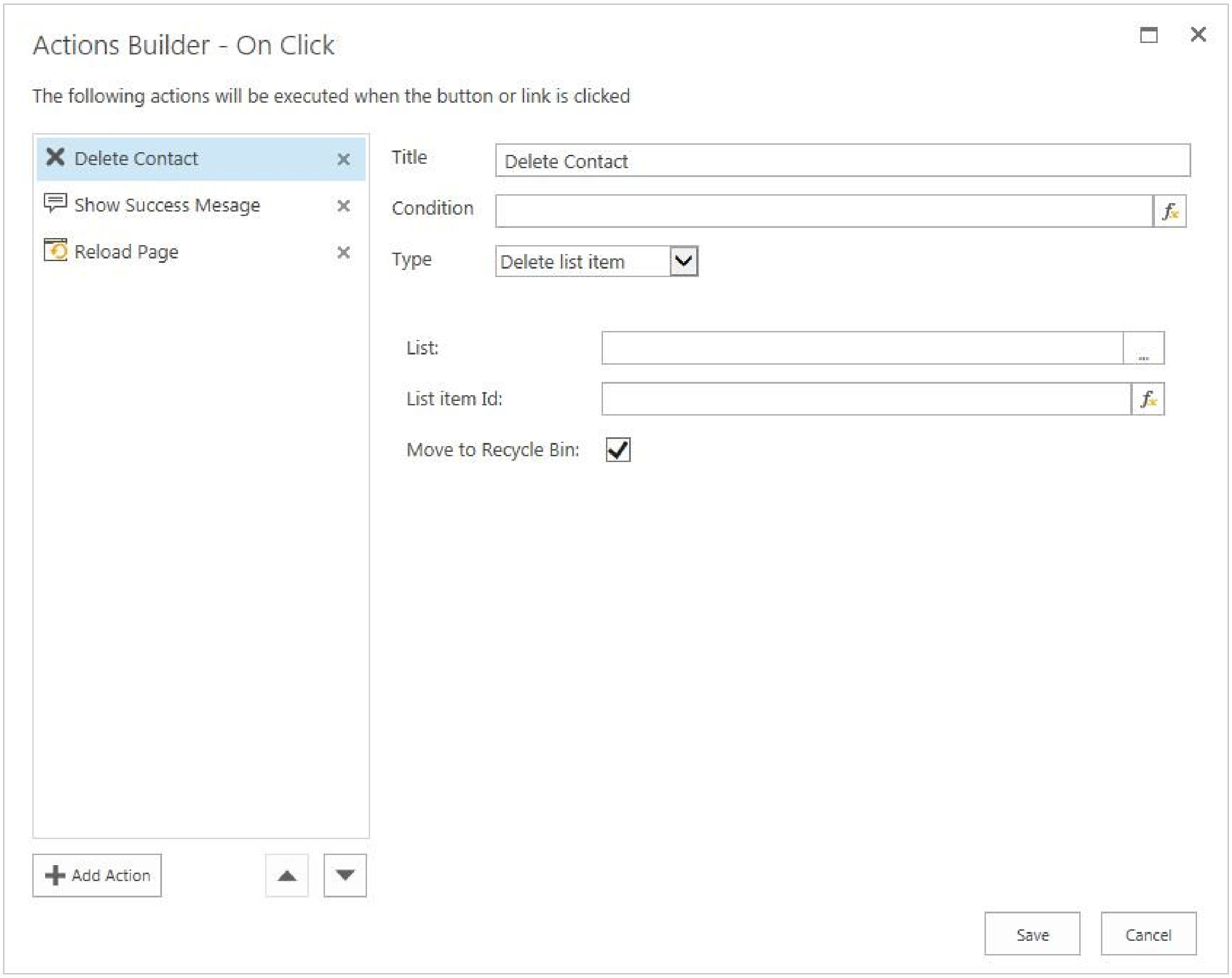
Actions Builder On Click

2.3.2. Delete List Item

Delete List Item action allows deleting list items dynamically.
When the action type “Delete List Item” is selected, the following additional properties can be specified:

- **List**: a list where an item should be deleted. It is possible to select the list from the current or any other site, by selecting the list in ‘Select List or Library’ dialog or typing the server relative URL to a list.
- **List Item Id**: ID or expression to be evaluated as ID of an item.
- **Move to Recycle Bin**: specifies if the items should be moved to the recycle bin and not deleted completely
  When the Delete List Item action is executed, the list item with the specified ID is moved to the recycle bin or deleted immediately depending on the configuration.

✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

- Delete Contact
- Show Success Mesage
- Reload Page

Title: Delete Contact

Condition:

Type: Delete list item

List:

List item Id:

Move to Recycle Bin: ☑

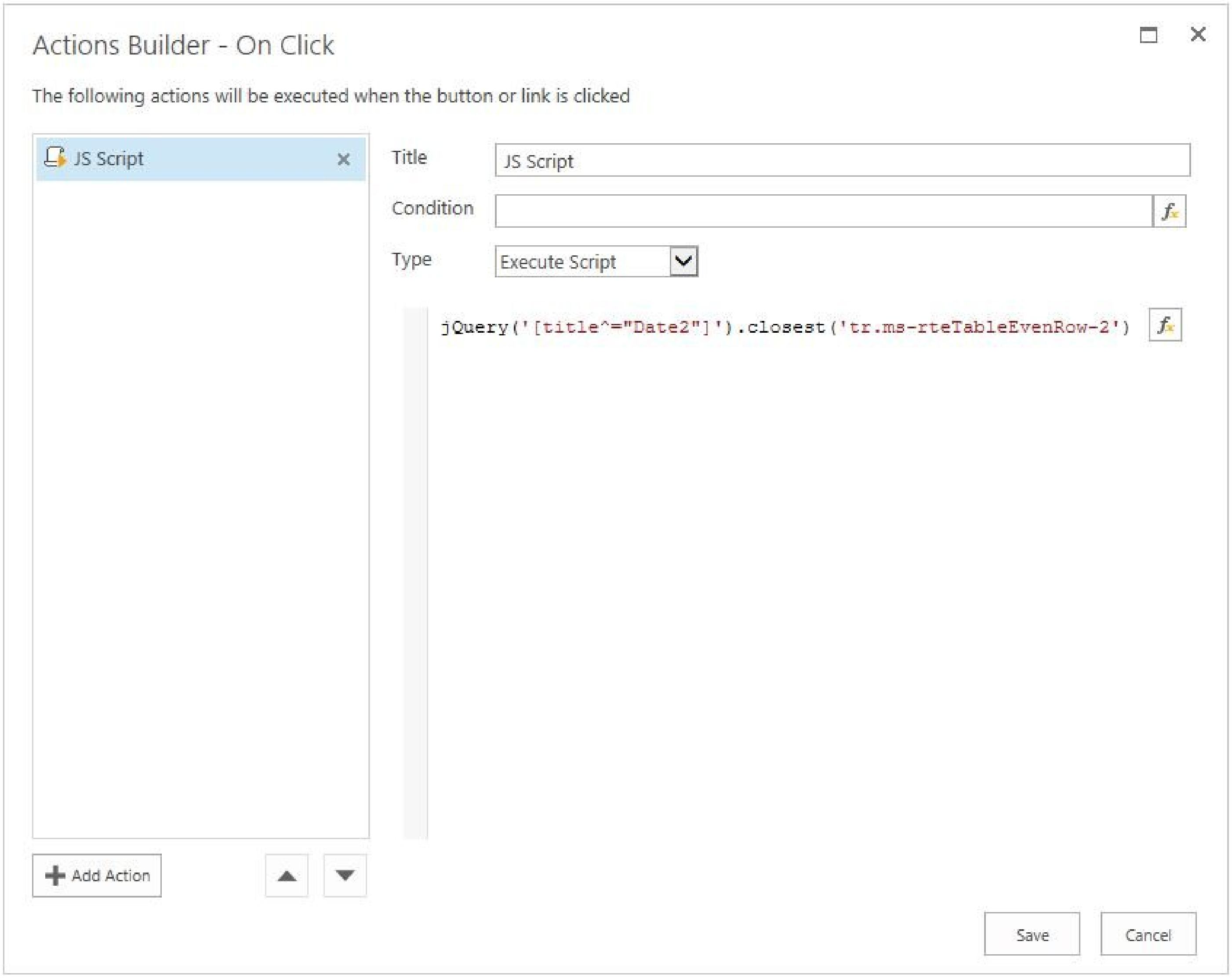
Delete List Item

2.3.3. Execute Script

Execute Script action allows executing JavaScript code.

When action type “Execute Script” is selected, the Script body property should be specified.

> Expressions are supported, thus making it possible e.g. to pass the current item ID to the script with [[ID]] inside.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: JS Script

Condition:

Type: Execute Script

```
jQuery('[title^="Date2"]').closest('tr.ms-rteTableEvenRow-2')
```

Actions Builder On Click

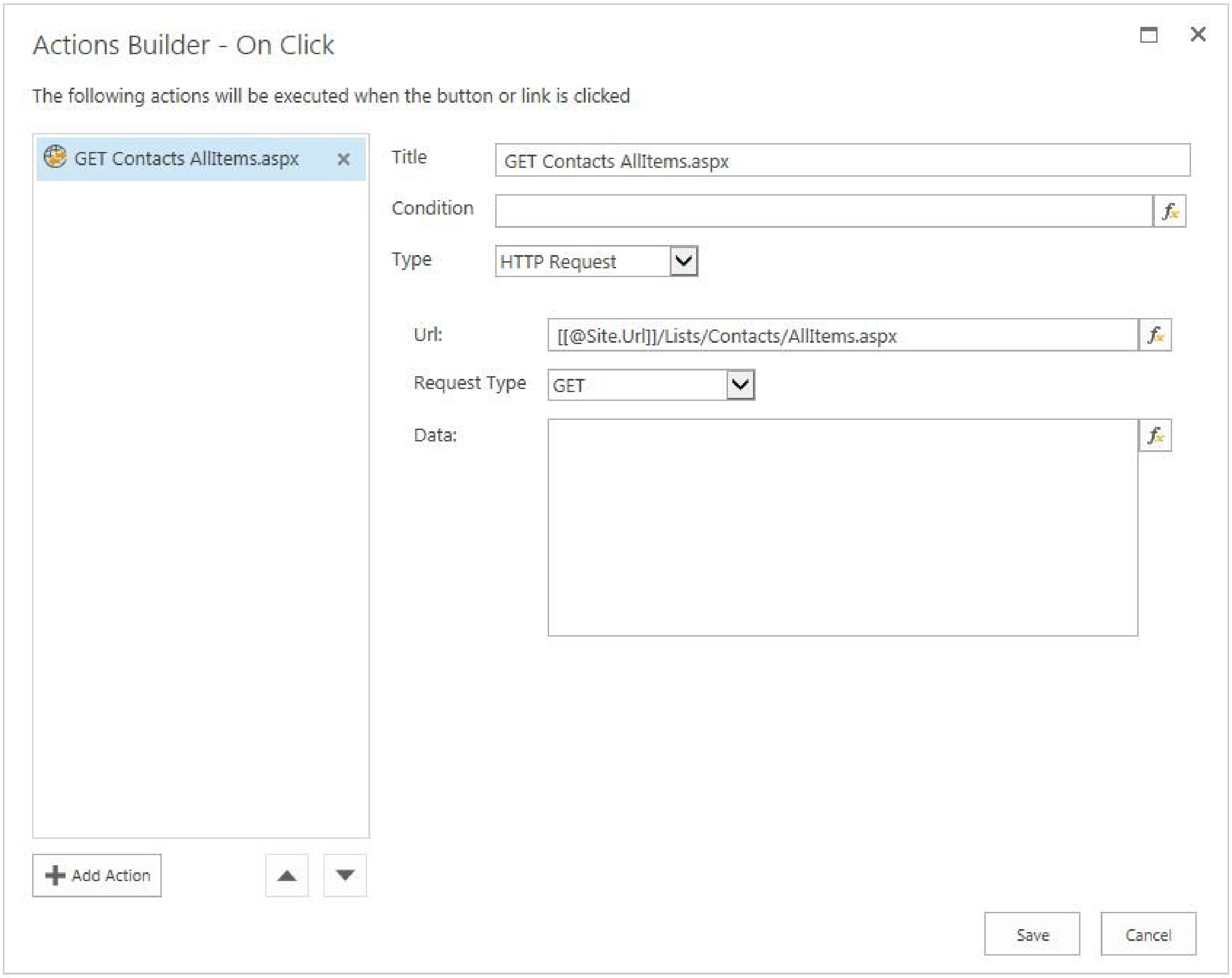
2.3.4. HTTP Request

HTTP Request action allows sending different types of requests:

- GET
- POST
- PUT
- DELETE
- JSONP

When action type “HTTP Request” is selected, the Data property can be specified. It is data to be sent to the server and is appended to the url or sent in the request body depending on the type of request. Object must be Key/Value pairs ({ name: “John”, location: “Boston” }) or an XML document.

✻ Expressions are supported, thus making it possible e.g. to pass the current item ID to the script with [[ID]] inside.


Actions Builder - On Click

The following actions will be executed when the button or link is clicked

GET Contacts AllItems.aspx

Title: GET Contacts AllItems.aspx

Condition:

Type: HTTP Request

Url: [[@Site.Url]]/Lists/Contacts/AllItems.aspx

Request Type: GET

Data:

HTTP Request

2.3.5. Open List Form

Open List Form action allows opening a New, Edit or Display form dynamically in a popup, dialog, the same or a new window.

When action type “Open List Form” is selected, the following additional properties can be specified:

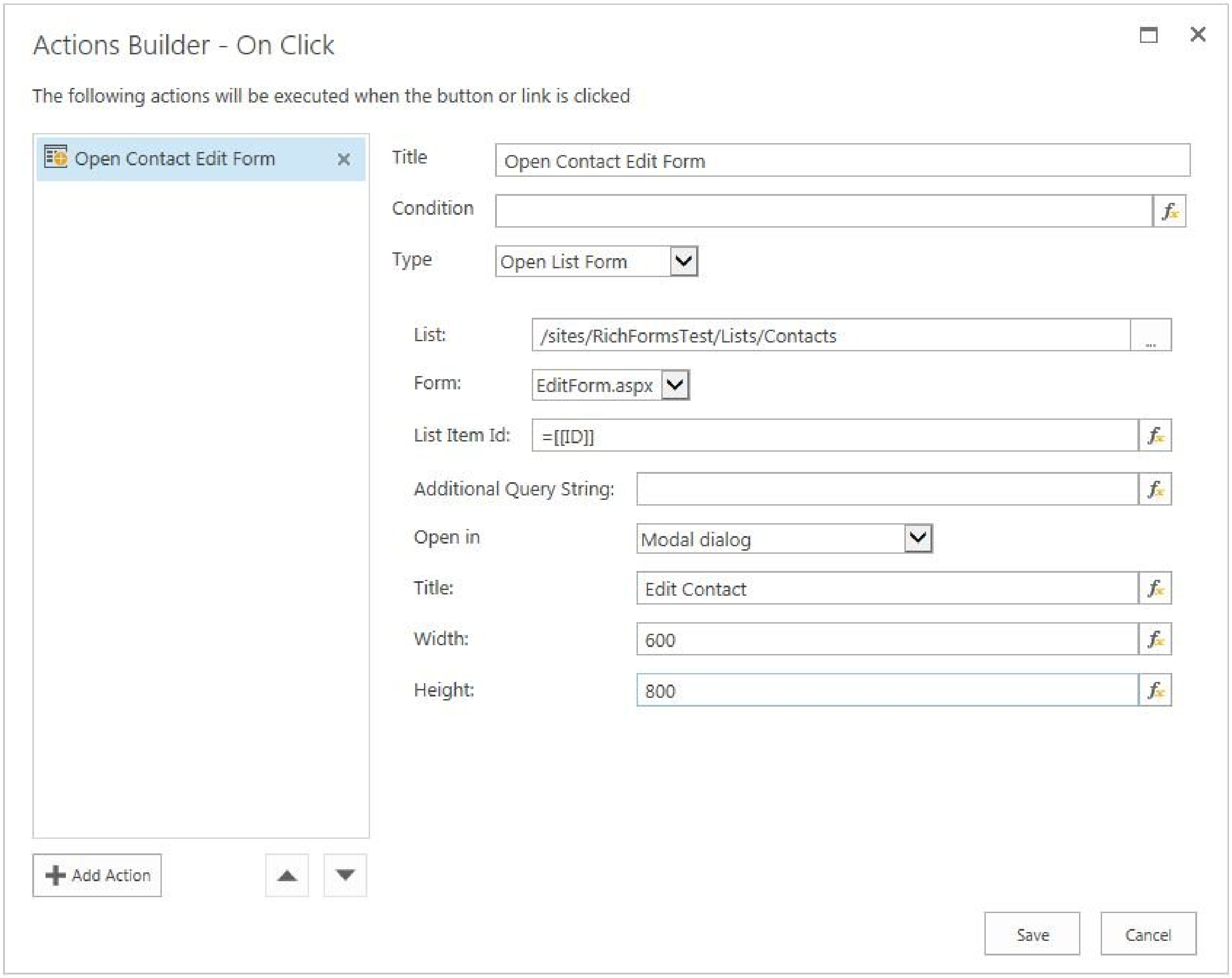
1. List: The list where the form is located. It is possible to select the list from the current or any other site, by selecting the list in ‘Select List or Library’ dialog or typing server relative URL to a list.
2. Form: Drop-down list with form types: New, Edit or Display (if a library is selected as the list, only the Edit and Display options are available).
   According to selected form type, the following fields appears:

- List Item Id: ID or expression to be evaluated as ID of an item if a Display or Edit form is selected.
- Content Type: Provides option for choosing content type to be added if a New form is selected.

1. Additional Query String: Allows adding further text to query string (the question mark “?” or ampersand “&” does not need to be typed at the beginning).
2. Open in: Provides options for choosing how a form should be opened. There are four options available:

- Modal dialog: the user is able to specify Title, Width (600 pixels by default) and Height (800 pixels by default) as well.
- Parent frame
- New window
- Popup browser window: the user is able to specify Width (600 pixels by default) and Height (800 pixels by default) as well.

✻ Expressions are supported, thus making it possible e.g. to pass the current item ID using the [[ID]] placeholder. You can use Expression Builder to easily assemble your expression.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: Open Contact Edit Form

Condition:

Type: Open List Form

List: /sites/RichFormsTest/Lists/Contacts

Form: EditForm.aspx

List Item Id: =[[ID]]

Additional Query String:

Open in: Modal dialog

Title: Edit Contact

Width: 600

Height: 800

Modal Dialog

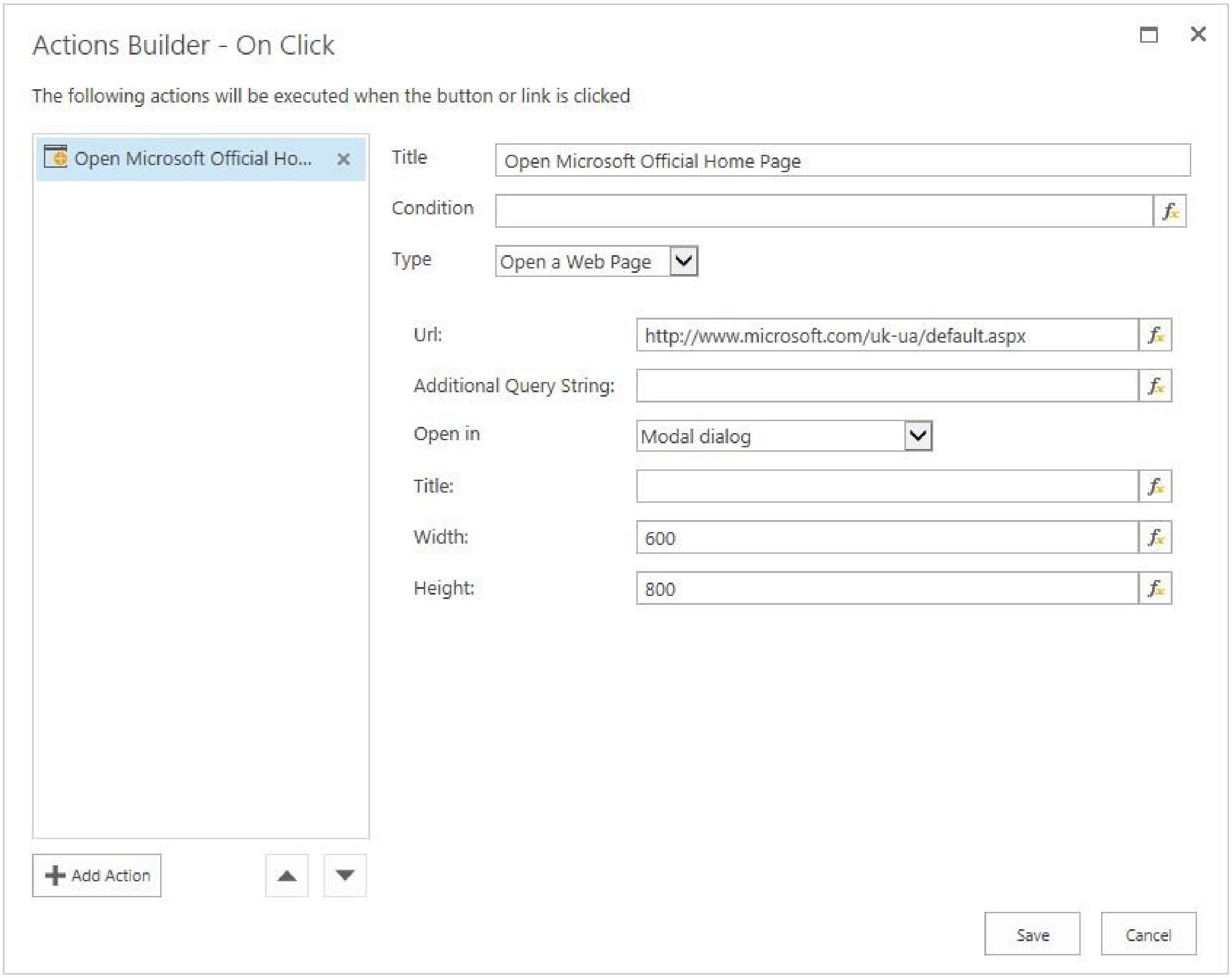
2.3.6. Open Web Page

Open Web Page action allows executing URL or expressions to be evaluated as URL for the link.

When action type “Open Web Page” is selected, the following additional properties should be specified:

1. Additional Query String: Allows adding further text to query string (the question mark “?” or ampersand “&” does not need to be typed at the beginning).
2. Open in: Provides options for choosing how a form should be opened. There are four options available:
    - Modal dialog: the user is able to specify Title, Width (600 pixels by default) and Height (800 pixels by default) as well.
    - Parent frame
    - New window
    - Popup browser window: the user is able to specify Width (600 pixels by default) and Height (800 pixels by default) as well.

✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Open Microsoft Official Ho...

Title: Open Microsoft Official Home Page

Condition:

Type: Open a Web Page

Url: http://www.microsoft.com/uk-ua/default.aspx

Additional Query String:

Open in: Modal dialog

Title:

Width: 600

Height: 800

Open a Web Page

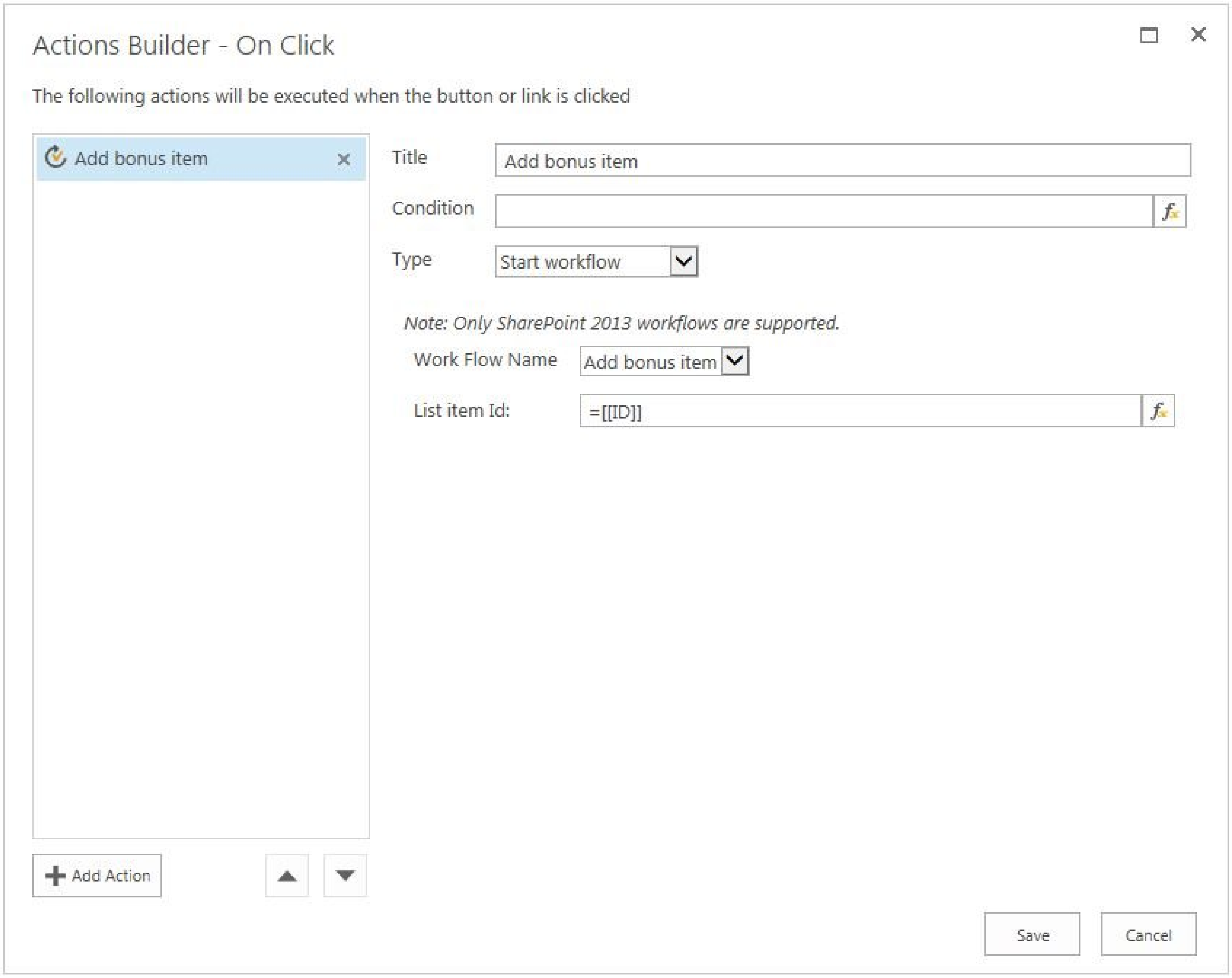
2.3.7. Redirect to URL

Start Workflow action allows starting a SharePoint Workflow dynamically. When action type “Start Workflow” is chosen, the following additional properties can be specified:

- Work Flow Name: The name of workflow for the current list or content type.
- List Item Id: ID or expression to be evaluated as ID of an item.

✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder.

✻ Only SharePoint 2013 workflows are supported.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: Add bonus item

Condition:

Type: Start workflow

Note: Only SharePoint 2013 workflows are supported.

Work Flow Name: Add bonus item

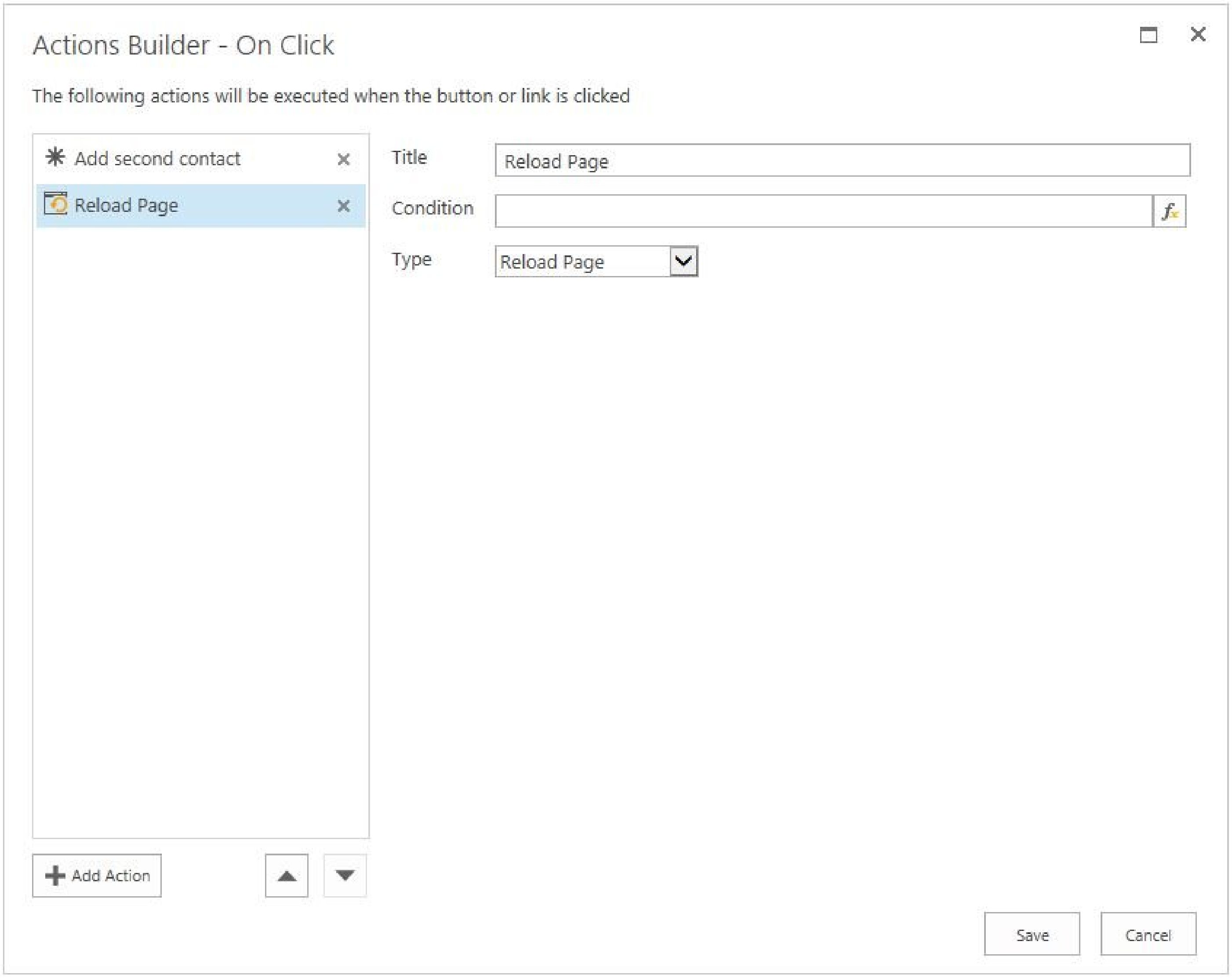
List item Id: =[[ID]]

Start Workflow

2.3.8. Reload Page

In some scenarios after some actions (i.e. updating or adding list item) it is necessary to reload the page. This can be done with the Reload Page action. It does not require any additional properties.

✻ “Reload Page” action should be the last in the actions list. All following actions will be ignored as it reloads the page.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

- Add second contact
- Reload Page

Title: Reload Page

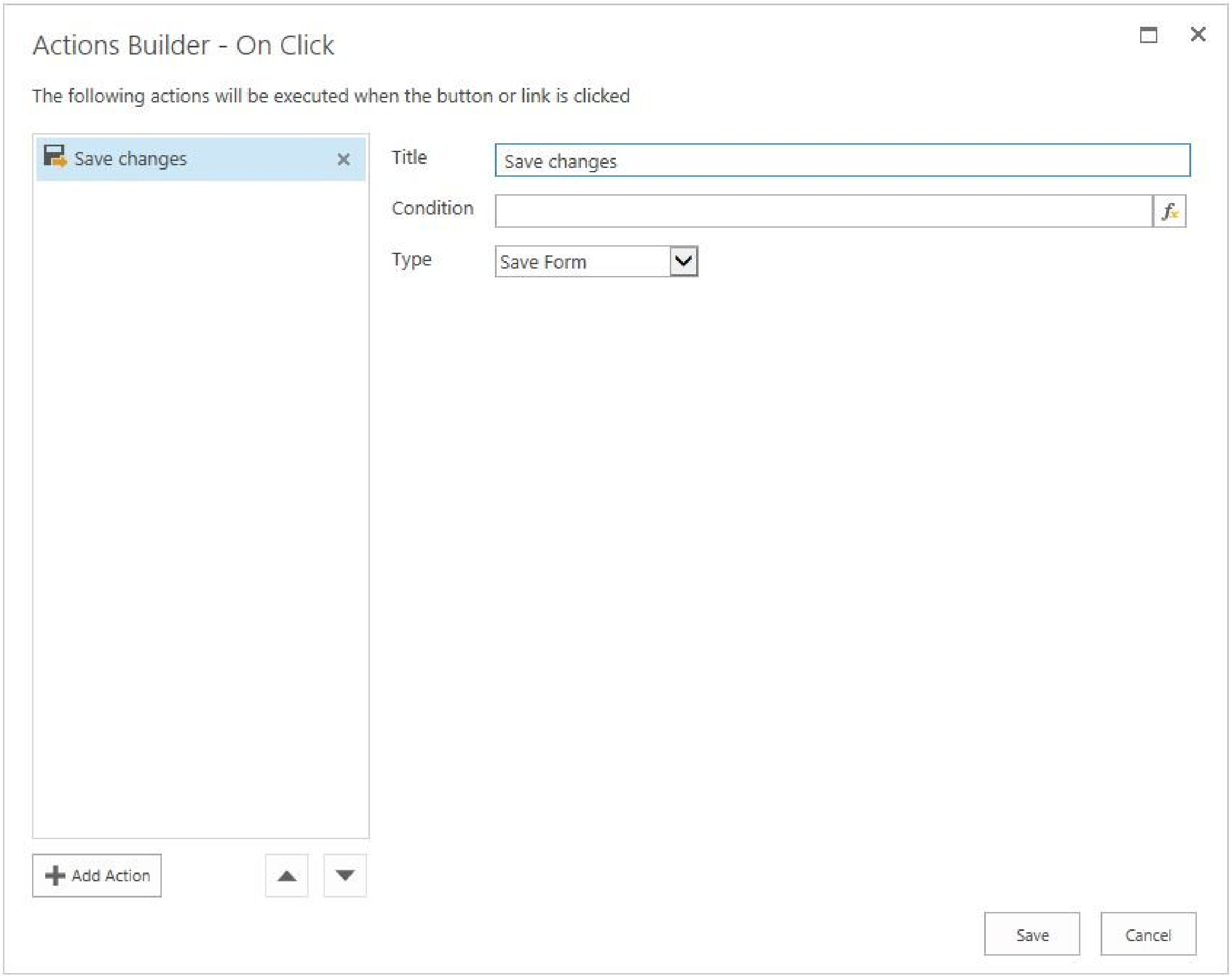
Condition:

Type: Reload Page

Reload Page

2.3.9. Save Form

Save Form action stores all modifications in the list form. It does not require any additional properties.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: Save changes

Condition:

Type: Save Form

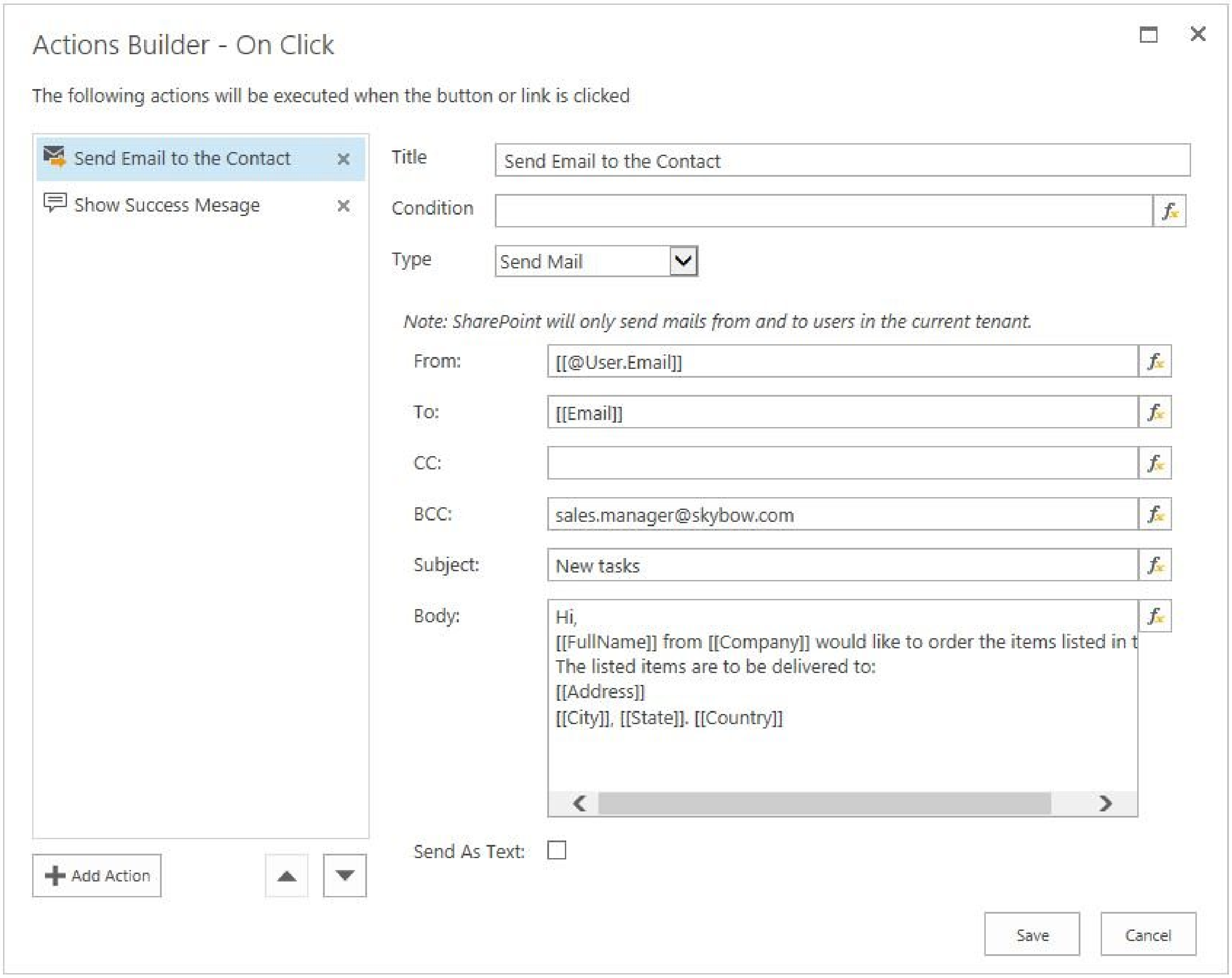
Save Form

2.3.10. Send Mail

Send Mail action allows sending dynamically generated emails from your site.
When action type “Send Mail” is selected, the following additional properties can be specified:

- **From**: An email address or an expression to be evaluated as an email address to be used as the sender.
- **To**: Email addresses or an expression to be evaluated as the email addresses to be used as receivers of the email. If multiple receivers are defined the email addresses must be separated by semicolons (i.e. “support@skybow.com; test@test.com”).
- **CC**: Email addresses or an expression to be evaluated as the email addresses to be used as “carbon copy” receivers of the email. If multiple receivers are defined the email addresses must be separated by semicolons (i.e. “support@skybow.com; test@test.com”).
- **BCC**: Email addresses or an expression to be evaluated as the email addresses to be used as “blind carbon copy” receivers of the email. If multiple receivers are defined the email addresses must be separated by semicolons (i.e. “support@skybow.com; test@test.com”).
- **Subject**: Subject of the email to be sent. Expressions can be used to dynamically generate this.
- **Body**: Actual email message to be sent. Expressions can be used to dynamically generate this.
- **Send As Text**: If enabled, email is sent in plain text otherwise it is sent as HTML mail.

✻ Mails will be sent only from and to users in the current tenant.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

- Send Email to the Contact
- Show Success Mesage

Title: Send Email to the Contact

Condition:

Type: Send Mail

Note: SharePoint will only send mails from and to users in the current tenant.

From: [[@User.Email]]

To: [[Email]]

CC:

BCC: sales.manager@skybow.com

Subject: New tasks

Body:

Hi,
[[FullName]] from [[Company]] would like to order the items listed in t
The listed items are to be delivered to:
[[Address]]
[[City]], [[State]]. [[Country]]

Send As Text: □

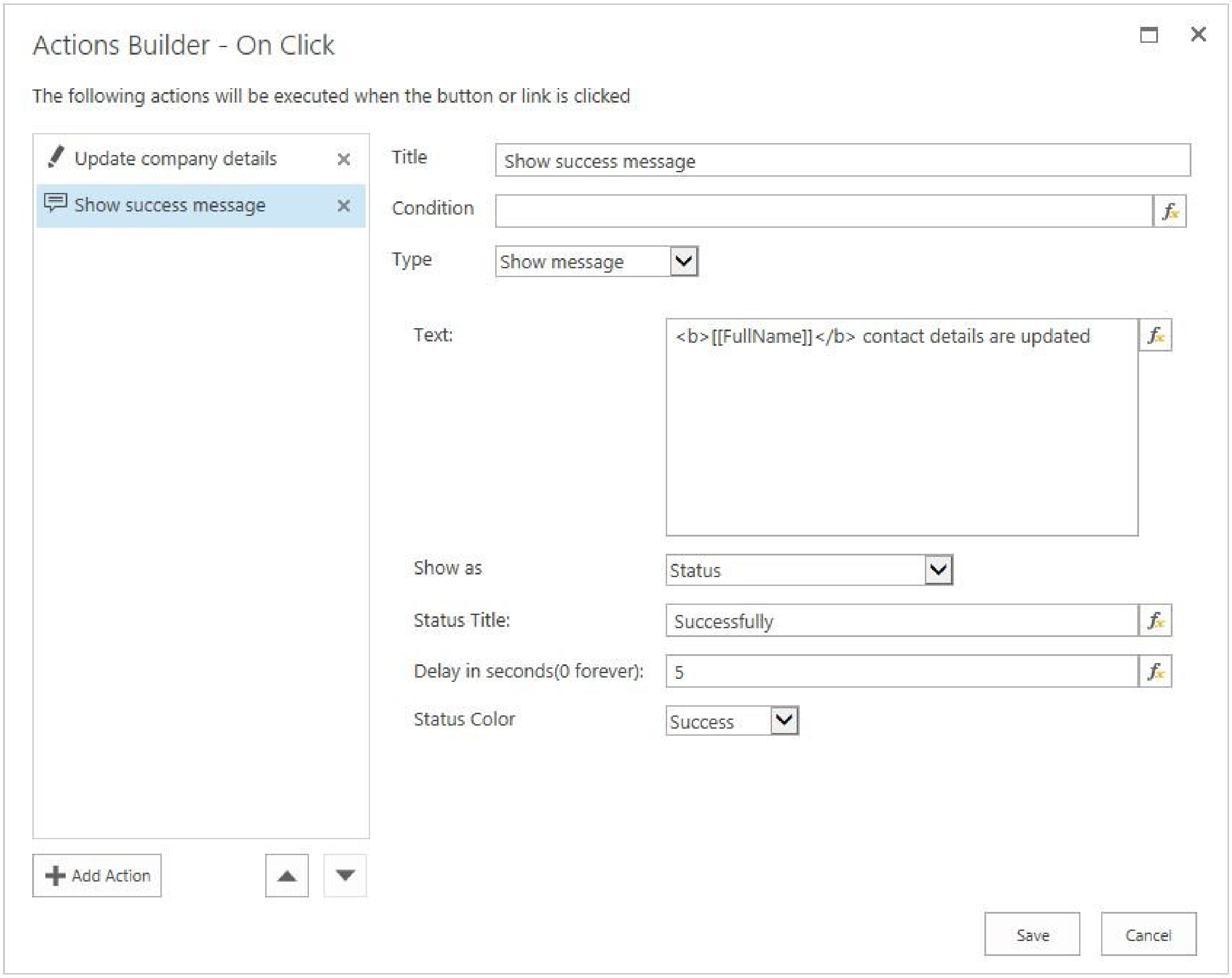
Send Email

2.3.11. Show Message

Show Message action allows showing message dynamically as a dialog, status or notification.
When action type "Show Message" is selected, the following additional properties can be specified:

1. **Text**: Template expression defining the message to be displayed (HTML tags can be used as well).
2. **Show as**: Provides options for choosing how a message should be shown. There are three options available:
    - **Dialog**: the user is able to specify Dialog Title, Width (400 pixels by default) and Height (150 pixels by default).
    - **Status**: the user is able to specify Status Title, Delay in seconds (if "0" is set, the message does not disappear until the page is refreshed), and Color (Information, Success, Warning, Error)
    - **Notification**: the user is able to specify Delay in seconds (if "0" is set, the message does not disappear until the page is refreshed).

> ✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: Show success message

Condition:

Type: Show message

Text: <b>[[FullName]]</b> contact details are updated

Show as: Status

Status Title: Successfully

Delay in seconds(0 forever): 5

Status Color: Success

Show Success Message

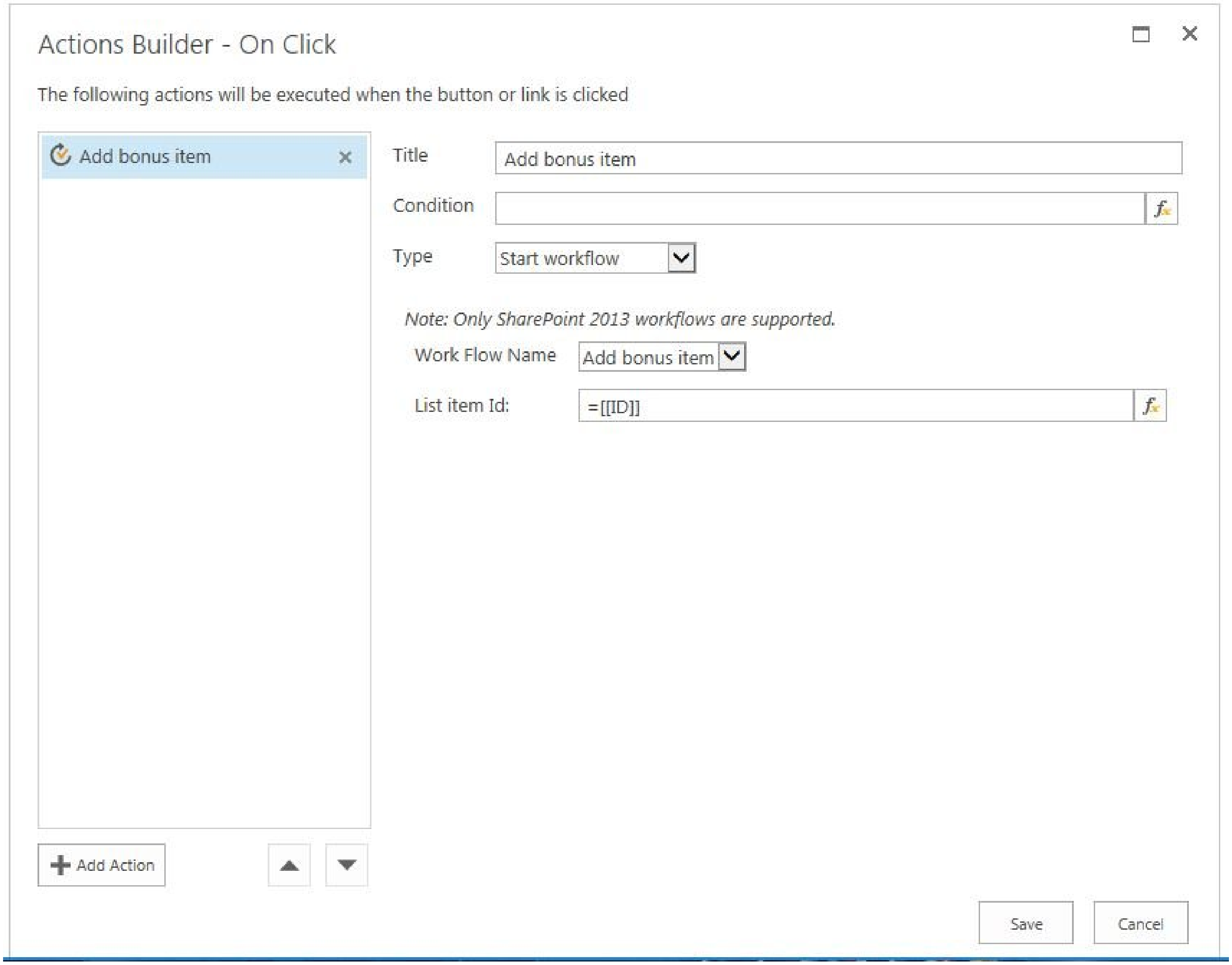
2.3.12. Start Workflow

Start Workflow action allows starting a SharePoint Workflow dynamically. When action type “Start Workflow” is chosen, the following additional properties can be specified:

- **Work Flow Name**: The name of workflow for the current list or content type.
- **List Item Id**: ID or expression to be evaluated as ID of an item.

> ✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder

> ✻ Only SharePoint 2013 workflows are supported.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

Title: Add bonus item

Condition:

Type: Start workflow

Note: Only SharePoint 2013 workflows are supported.

Work Flow Name: Add bonus item

List item Id: =[[ID]]

Start Workflow

2.3.13. Update List Item

Update List Item action allows updating a list item immediately with configurable values.

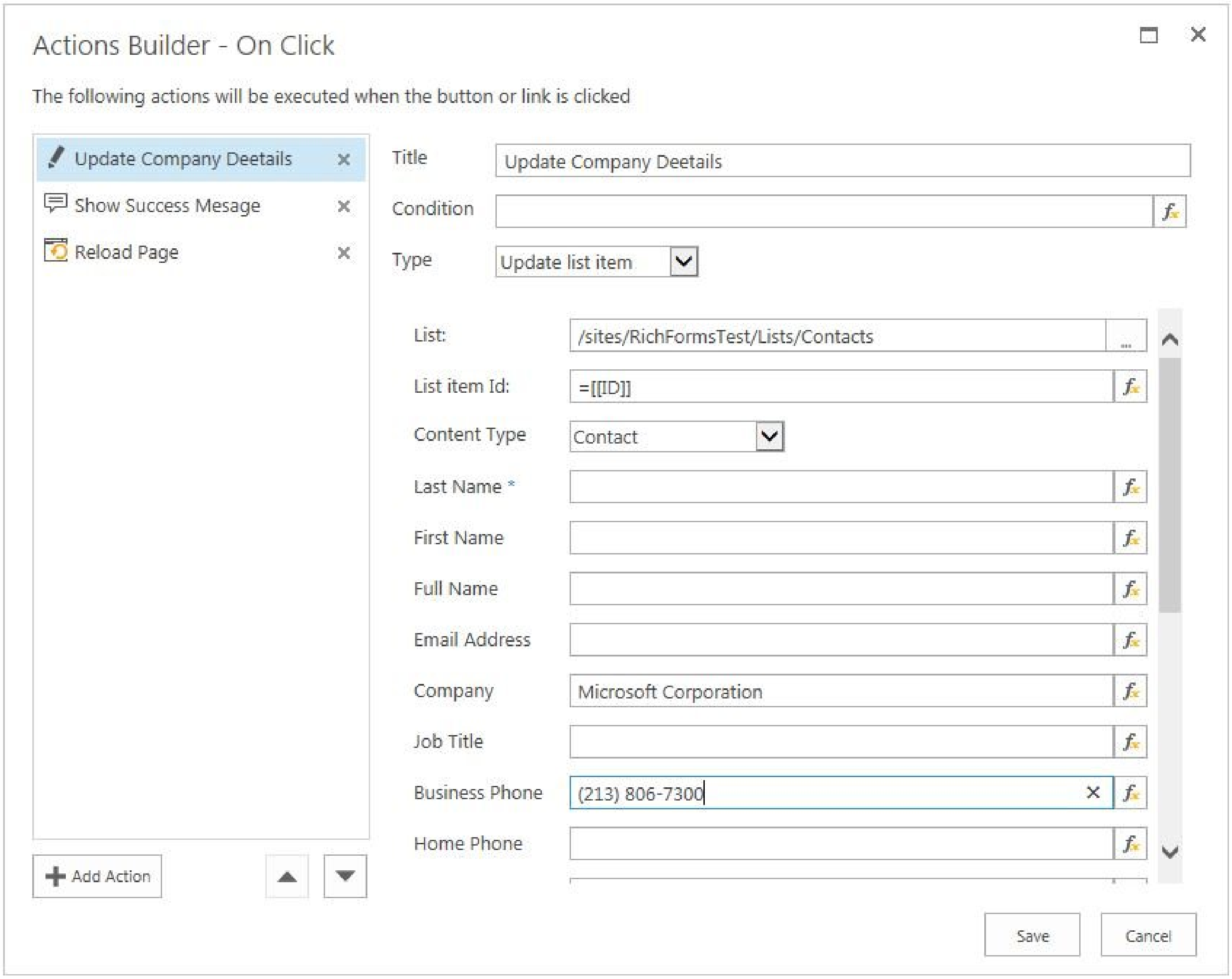
When action type “Update List Item” is selected, the following additional properties can be specified:

- List: a list where the list item should be updated. It is possible to select the list from the current or any other site, by selecting the list in ‘Select List or Library’ dialog or typing a server-relative URL to a list.
- List Item ID: ID or expression to be evaluated as ID of an item.
- Fields specific for current list: At the bottom of the configuration all fields of the selected list are shown, so that for every field the value to be set can be specified.

✻ Expression-based values for these fields can be easily set using the placeholders available in Expression Builder.

✻ It is possible to set null (as Assignment expression “=null”) and “ “ (empty string) values.

✻ Only specified fields will be updated. If you leave the field blank, it will not be changed.



Actions Builder - On Click

The following actions will be executed when the button or link is clicked

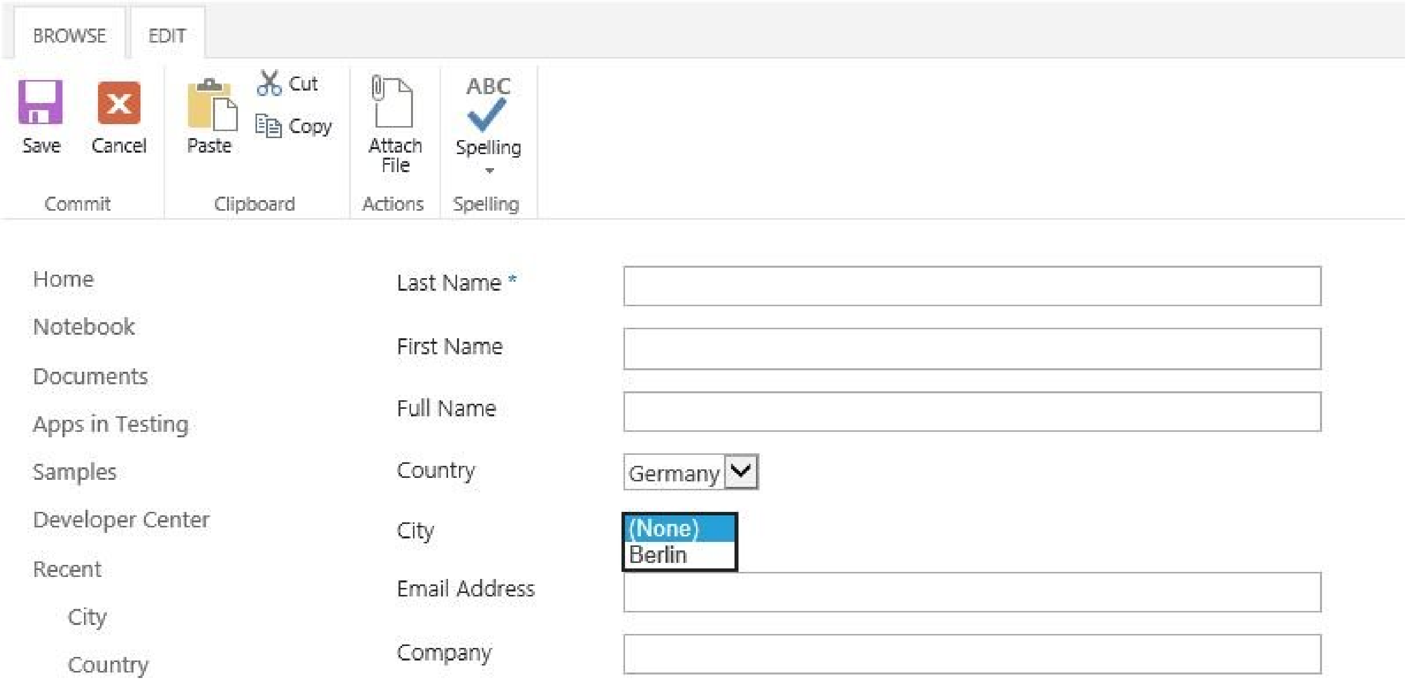
- Update Company Deetails
- Show Success Mesage
- Reload Page

Title	Update Company Deetails
Condition	
Type	Update list item
List:	/sites/RichFormsTest/Lists/Contacts
List item Id:	=[[ID]]
Content Type	Contact
Last Name *	
First Name	
Full Name	
Email Address	
Company	Microsoft Corporation
Job Title	
Business Phone	(213) 806-7300
Home Phone	

Update List Item

2.4. Lookup Filtering

Lookup filtering greatly enhances default SharePoint’s lookup functions. It reduces wait times as lookup results are filtered according to parameters that you set.
As well cascading lookups are now easy to implement using Filter functionality.



Last Name *

First Name

Full Name

Country Germany

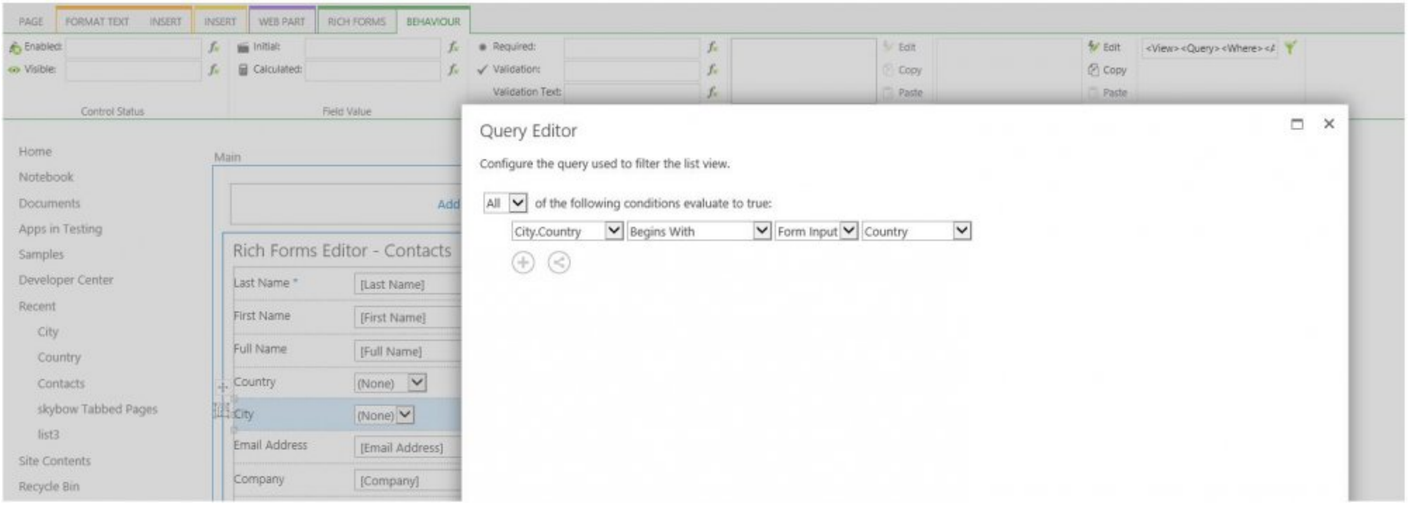
City (None)
Berlin

Email Address

Company

Lookup Filtering

Filter criteria can be set on **Filter** section of Behaviour tab when some lookup field is selected.
CAML Query Builder is accessible when Edit Query button is clicked.



Query Editor

Configure the query used to filter the list view.

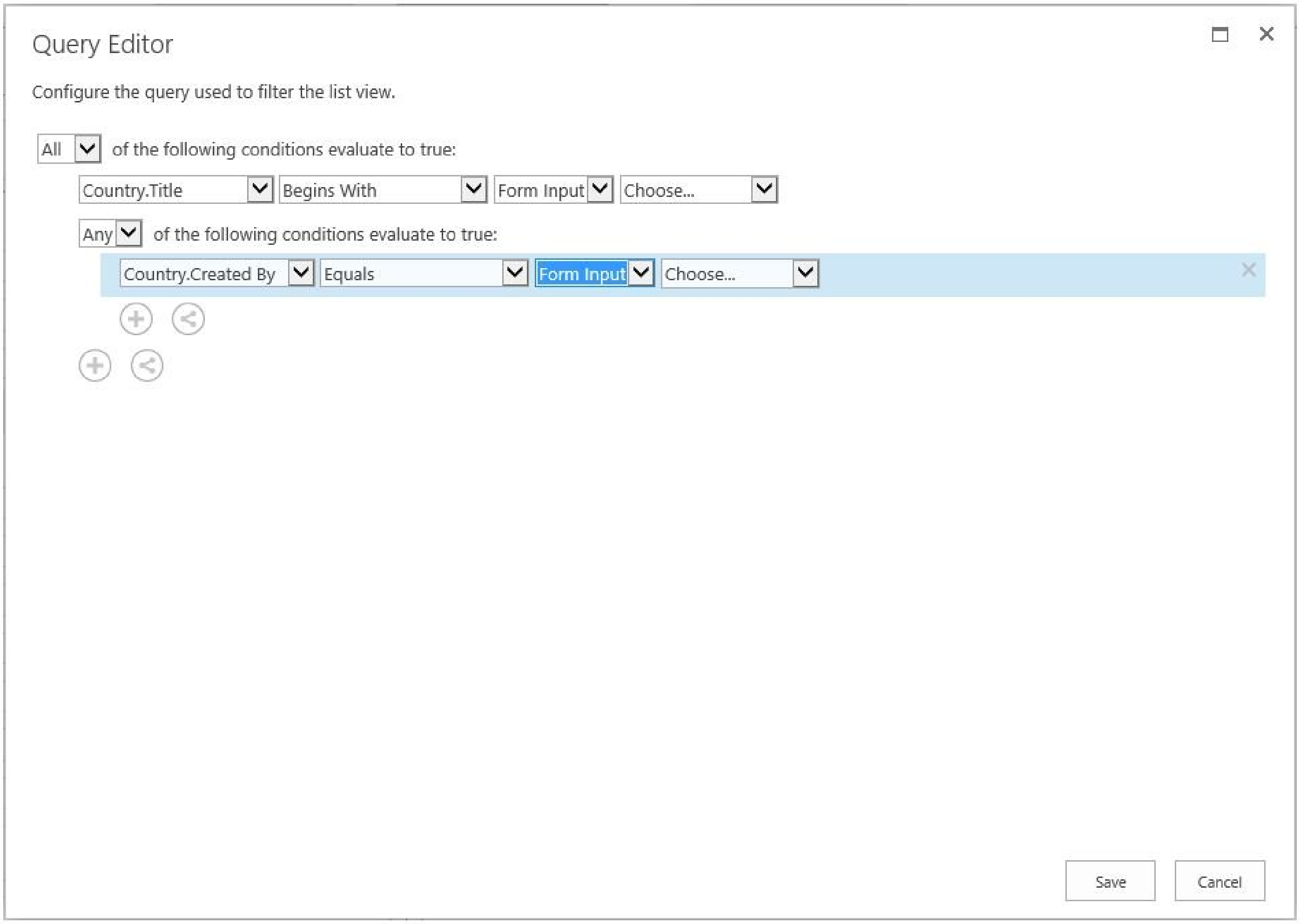
All of the following conditions evaluate to true:

City.Country | Begins With | Form Input | Country

Lookup Filtering

2.5. Query Editor

Complicate your queries quick and easy with CAML Query Editor.
Lightning Forms app uses a drag and drop interface. You can drag and drop one item, as well as a group with multiple items in it.



Query Editor

Configure the query used to filter the list view.

All of the following conditions evaluate to true:

Country.Title | Begins With | Form Input | Choose...

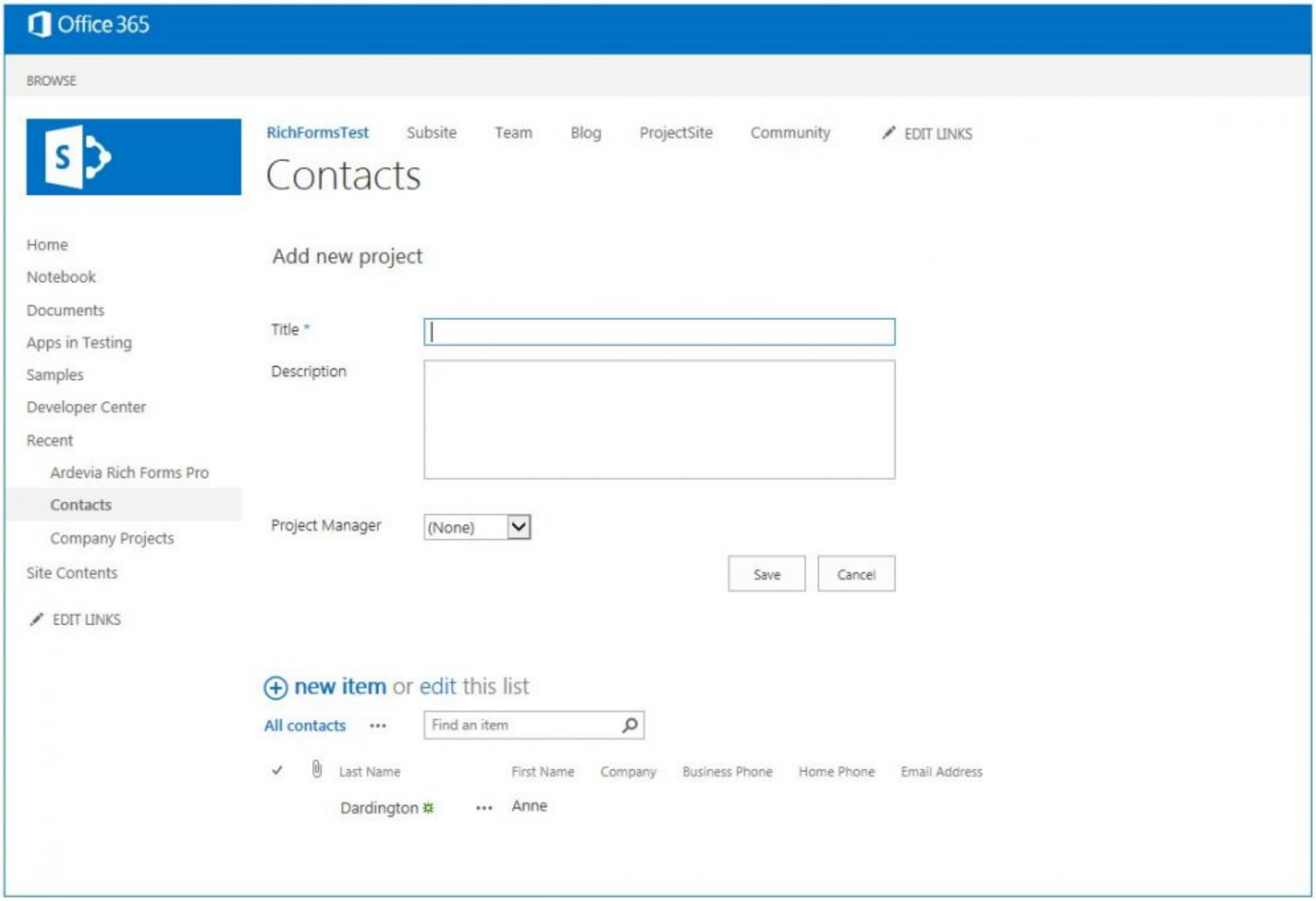
Any of the following conditions evaluate to true:

Country.Created By | Equals | Form Input | Choose...

Query Editor

2.6. Embed List Form App Part

Rich Forms provides a web part to embed a list form in any web page in your site collection.



Embed List Form Web Part

The app part provides the following configuration options:

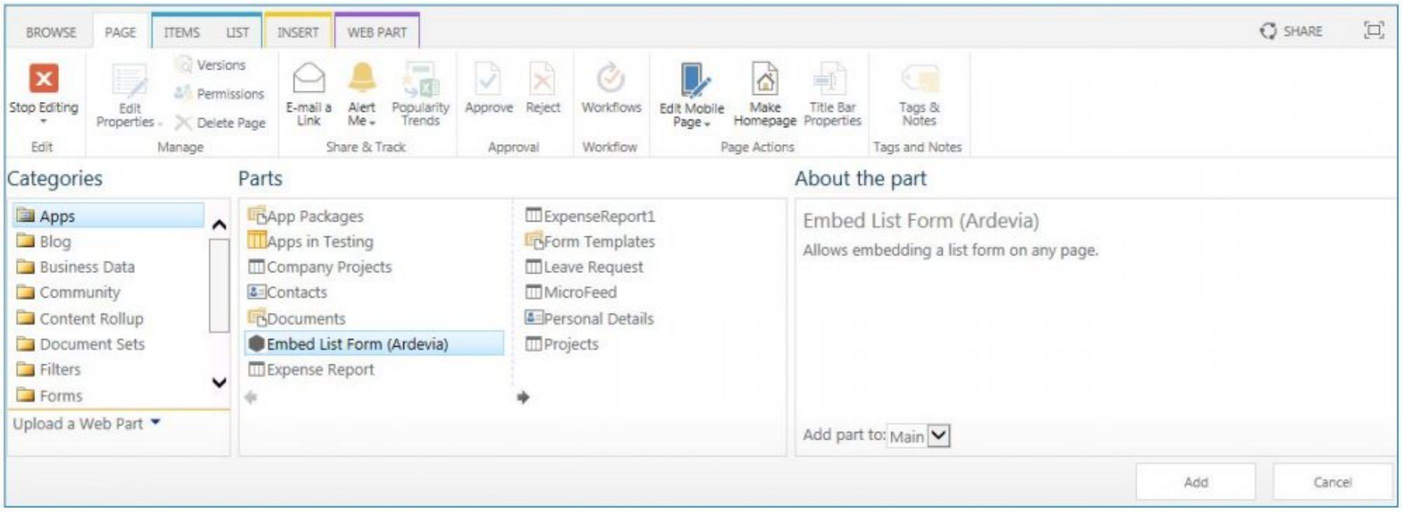
Select any list form from any site
Define the ID of the list item to show in edit or display forms
Show or hide the ribbon for the form (only for forms customized with Rich Forms)
Define fixed height and width of form (only if not customized with Rich Forms)
Define where to redirect on successfully saving the form
It also provides a link to directly go to customizing the selected list form.

To insert a list form on any page, follow these steps:

Add Embed List Form app part

Use the SharePoint functionality for adding an app part:

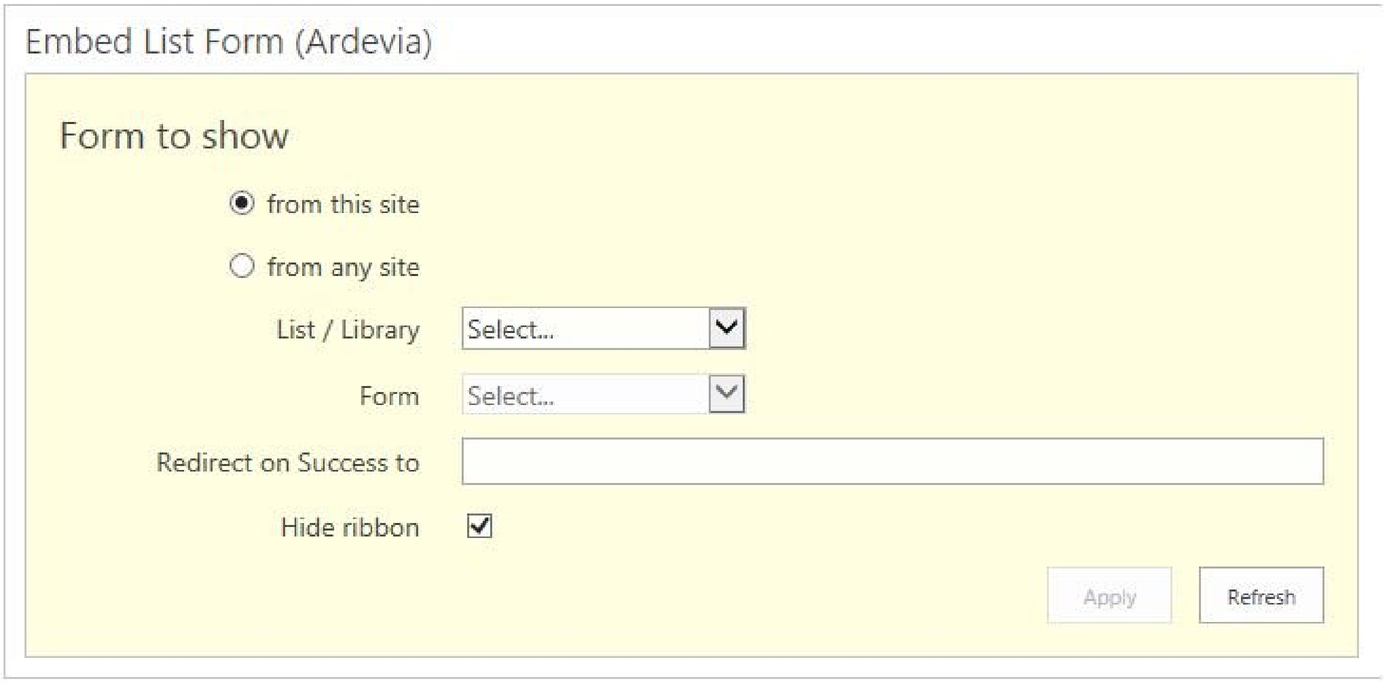
- Select the **Embed List Form**.
- Click Add to insert it.



Embed List Form

Configure List Form and App Part Properties

In the dropdowns, select the list or library and form to be displayed.



Embed List Form (Ardevia)

Form to show

● from this site

○ from any site

List / Library: Select...

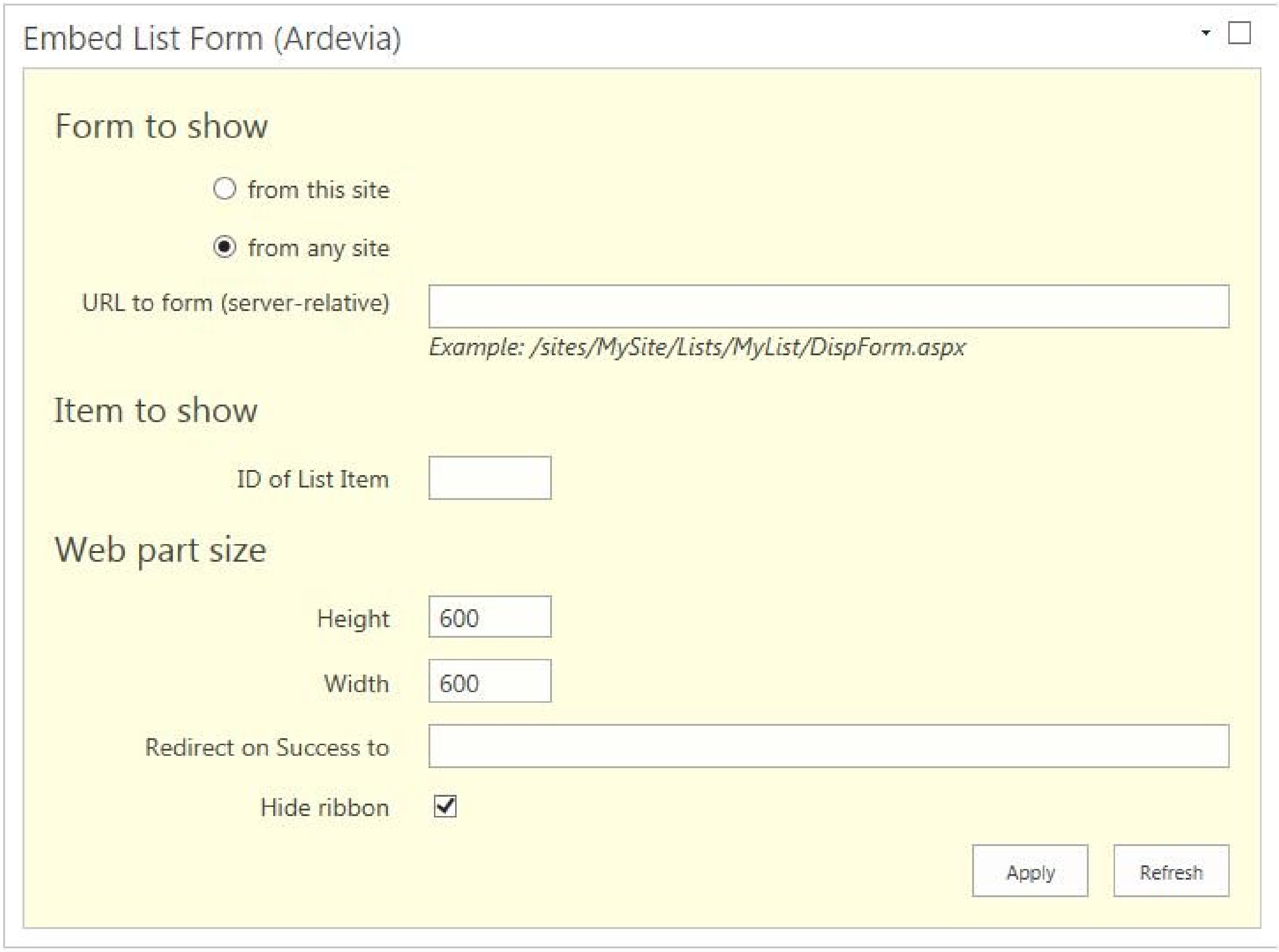
Form: Select...

Redirect on Success to:

Hide ribbon ☑

Configure List Form and App Part Properties

Embed List Form web part supports cross-site list selection. To use a list form from another site, set the server-relative URL in the corresponding input.



Embed List Form (Ardevia)

Form to show

○ from this site

● from any site

URL to form (server-relative)

Example: /sites/MySite/Lists/MyList/DispForm.aspx

Item to show

ID of List Item

Web part size

Height: 600

Width: 600

Redirect on Success to

Hide ribbon ☑

Cross Site List Collection

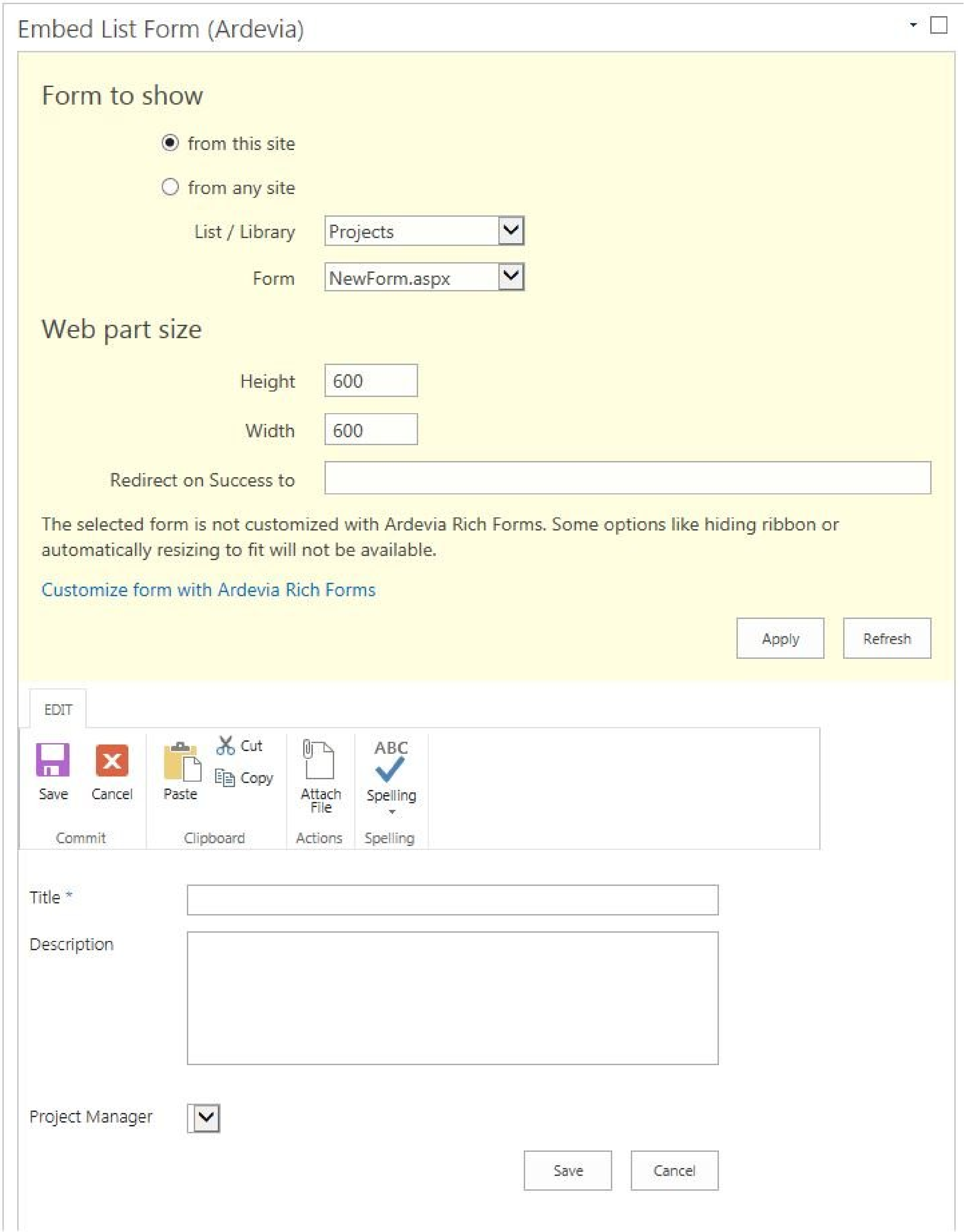
- Set the ID of List Item, if Display or Edit forms are selected.
- Configure the app parts height and width. This will only be available and necessary if the selected form has not been customized using Rich Forms. Forms already customized will resize correctly automatically.
- Set the Redirect on Success to URL where the user should be redirected to after saving.

If Redirect on Success to property is not set, the page will be refreshed (in case when the list is from the current site) or saving will navigate to the list view page (in case when the list is from another site).

Redirection to external sites is possible for customized forms.

- Uncheck Hide ribbon option, if you want ribbon to be displayed in the form. This only works for forms already customized with Rich Forms.

> ✻ Uncustomized form always contains out of the box form ribbon.



Embed List Form (Ardevia)

Form to show

● from this site

○ from any site

List / Library: Projects

Form: NewForm.aspx

Web part size

Height: 600

Width: 600

Redirect on Success to:

The selected form is not customized with Ardevia Rich Forms. Some options like hiding ribbon or automatically resizing to fit will not be available.

Customize form with Ardevia Rich Forms

Title *

Description

Project Manager

Customize List Forms

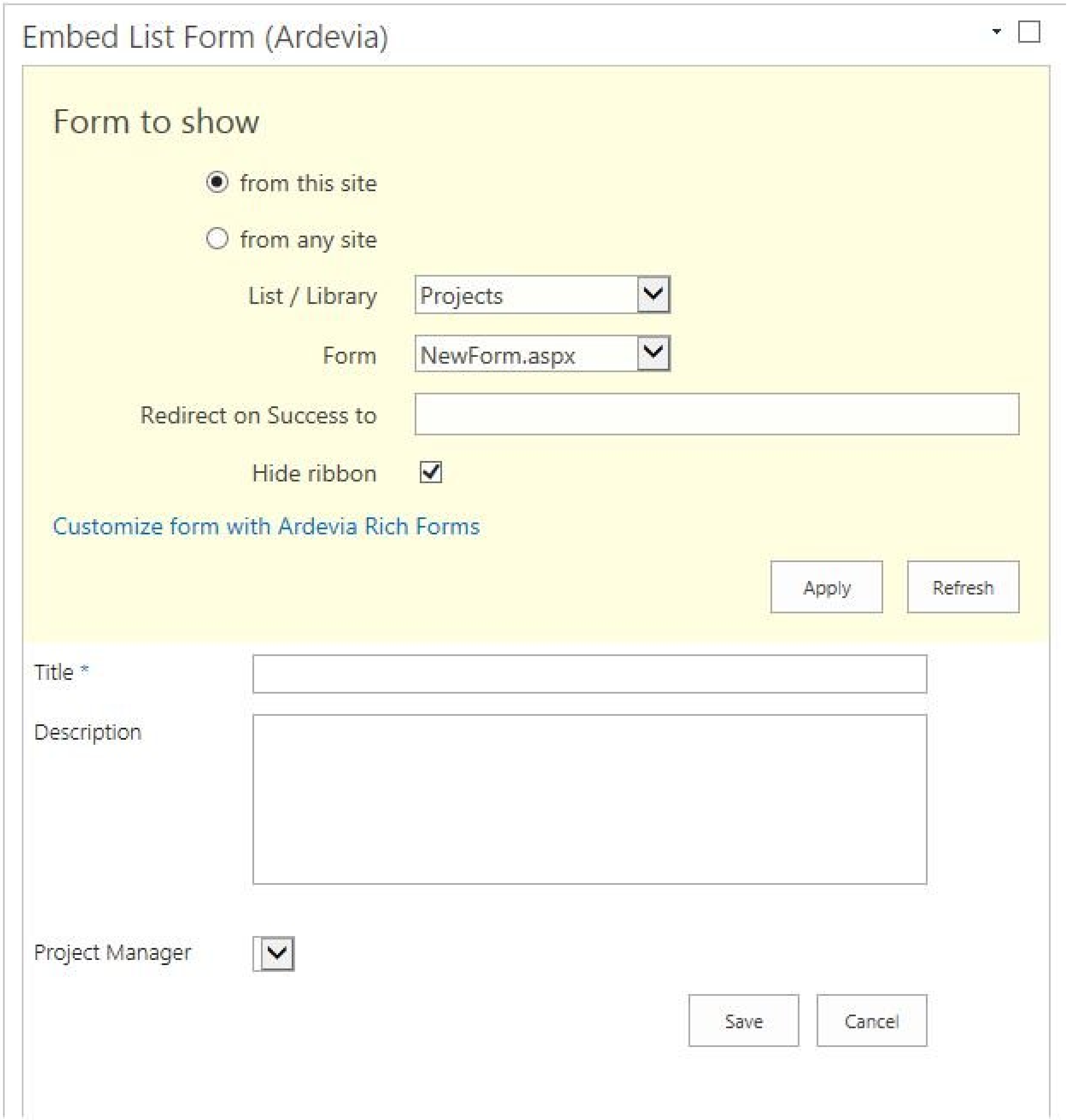
Customize, refresh and save configuration

Clicking on the **Customize form with Lightning Forms** link navigates to the form page in design mode.

Edit the form in the Rich Forms editor
Save the changes.

After clicking the **Refresh** button the customized form will be displayed.

Clicking on the **Apply** button saves the configuration in the app part’s properties.



Embed List Form (Ardevia)

Form to show

● from this site

○ from any site

List / Library: Projects

Form: NewForm.aspx

Redirect on Success to:

Hide ribbon ☑

Customize form with Ardevia Rich Forms

Title *

Description

Project Manager

3. Tutorials