Form Builder - User Guide

Orckestra 2017-02-10

C1 CMS

Contents

C1 CMS

1   Introduction

The Form Builder allows you to quickly and easily create well-designed, mobile-friendly and accessible web forms and insert them on pages.
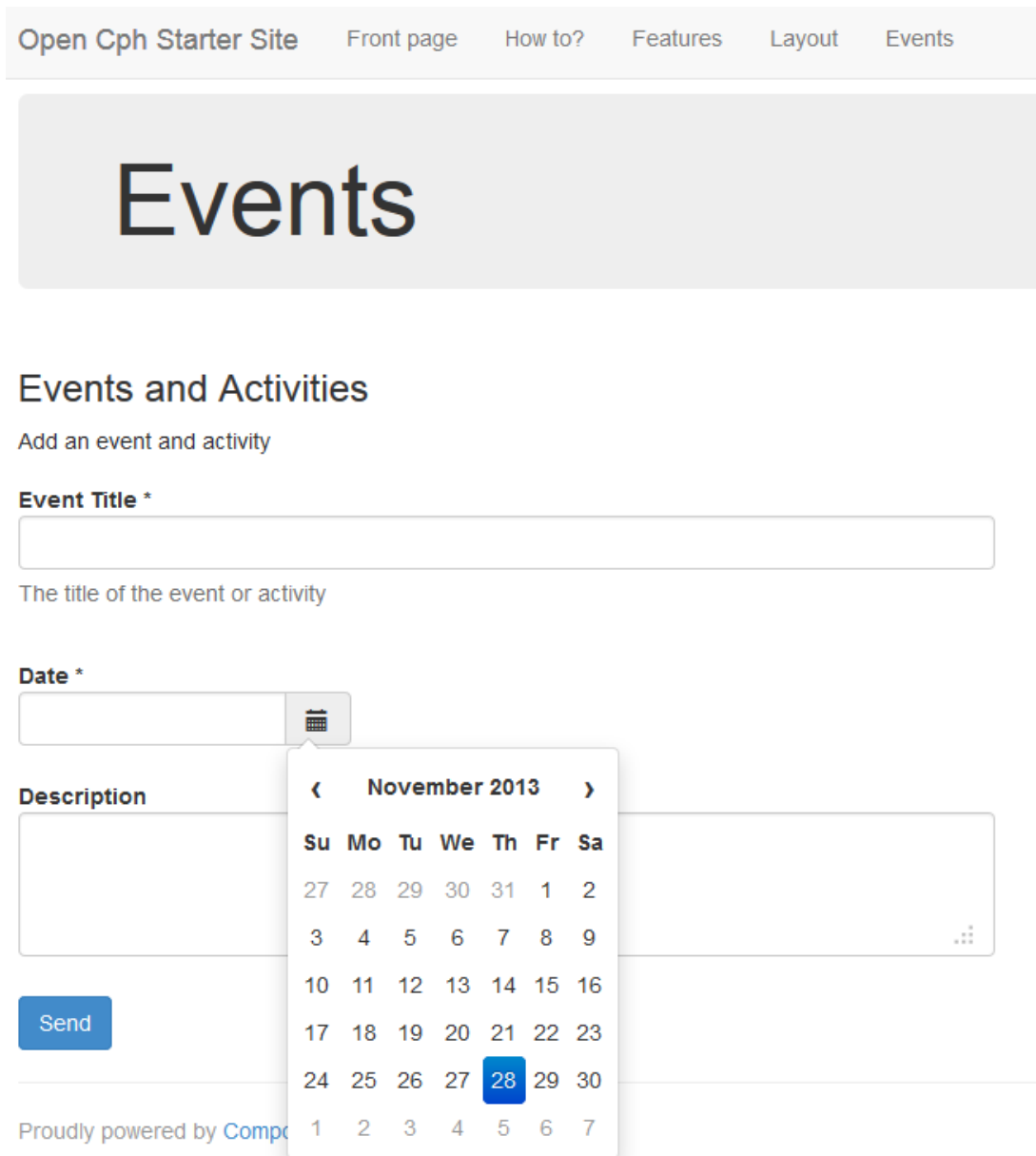

Figure 1: A form created with the Form Builder

Creating a form is as simple as dragging visual components in the Form Builder window and dropping them where you need them on the form.

Each component that serves as a field on the resulting form, comes with reasonable defaults but you can fine-tune each field by setting their properties while editing the form.
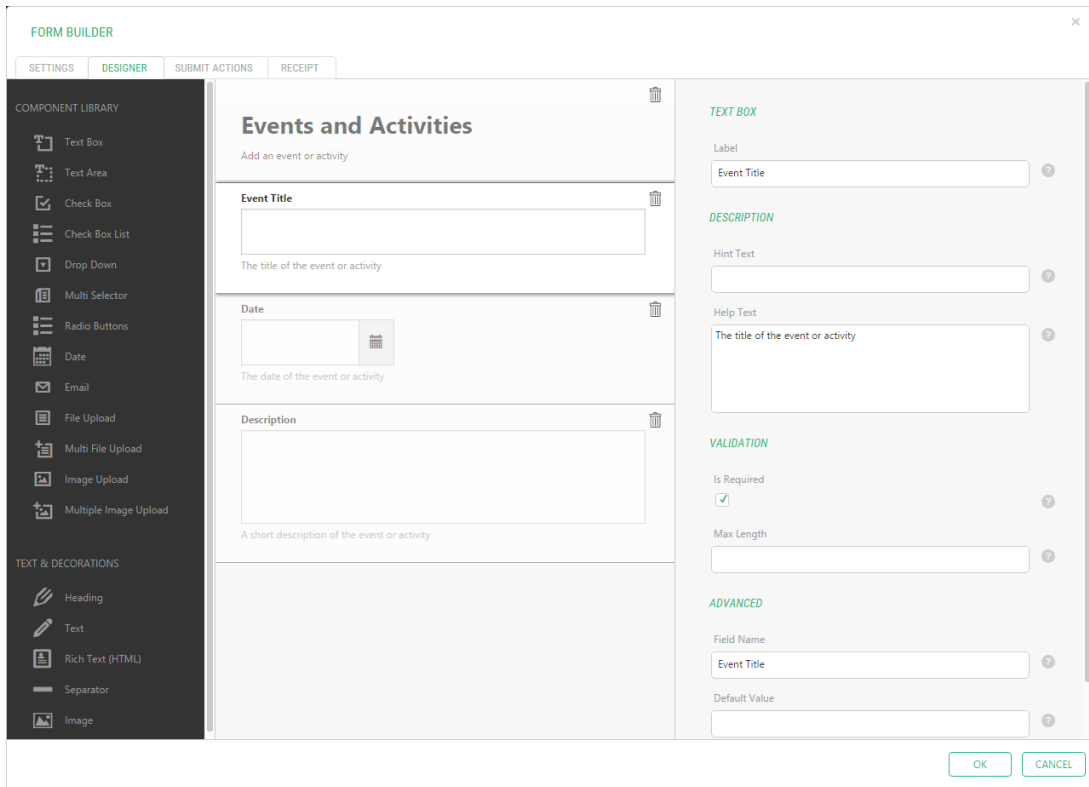
Figure 2: Creating a form in the Form Builder

By default the data submitted with the form is saved on the website. You can browse it in the CMS Console as well as download it as an XML file or export it to a Microsoft Excel spreadsheet.

Figure 3: Viewing submitted data

You can also configure the form to send the submitted data to a designated recipient by email.

The Form Builder provides you with a set of field, text and decoration components out-of-the-box enough to start building forms immediately. You can however add custom fields to the Component Library or reuse the existing ones with new settings if necessary.

Besides, you can integrate your forms with custom back-end systems such as a CRM.

The forms created with Form Builder are all based on Bootstrap 3. You can customize their "look and feel" by HTML and CSS behind them.

Note. Extending the Form Builder with custom fields and components, integrating forms with custom back-end systems and customizing their appearance and behavior as well as localizing the forms are topics of the "*Form Builder Developer Guide*".

1.1   Who Should Read This Guide?

This guide is mostly intended for editors or designers who need to create forms and insert them on pages.

You will primarily work within the Content perspective, editing pages and working with submitted data. You should know how to work with pages in C1 CMS, namely, insert CMS functions.

You may also need access to the Media perspective if your forms include images from Media archive.

1.2   Getting Started

To get started with the Form Builder, consider learning to do the following activities.

Getting Started		
Step	**Activity**	**Chapter or section**
1	Insert a form on a page	Inserting New Forms
2	Edit a form	Editing Forms
3	Add fields to a form	Adding Fields to Forms
4	Edit field properties	Editing Field Properties
5	View submitted data	Viewing Submitted Data
6	Set up form submission by email	Receiving Submitted Data by Email

C1 CMS

2   Inserting New Forms

The Form Builder is available via a CMS function called
"Composite.Forms.FormBuilder.Form".

When you insert, or edit the properties of, this function on a page, a Form Builder window
opens. In this window you can drag and drop fields onto the form and configure its other
parameters.

To insert a new form on a page:

1. In the **Content** perspective, edit a page where you want to insert your form.
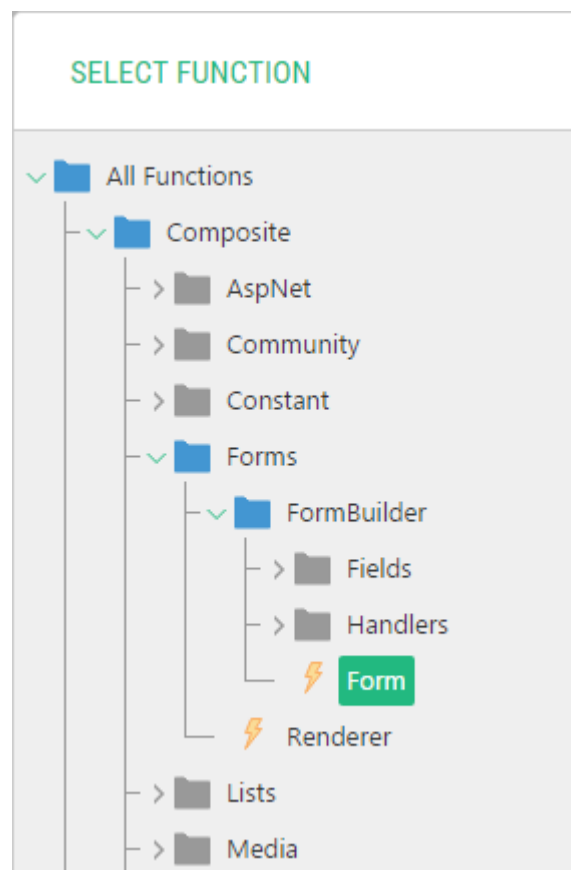2. Add the **Composite.Forms.FormBuilder.Form** function (**Insert** | **Function**).

Figure 4: Inserting a form

3. In the **Form Builder** window, on the **Settings** tab, specify the **Name** of the form and
   set other parameters here if needed. (These settings are discussed in detail in
   "Settings".)

Figure 5: Form's settings

4. On the **Designer** tab, add fields and other components and set their properties. (See "Adding Fields to Forms")
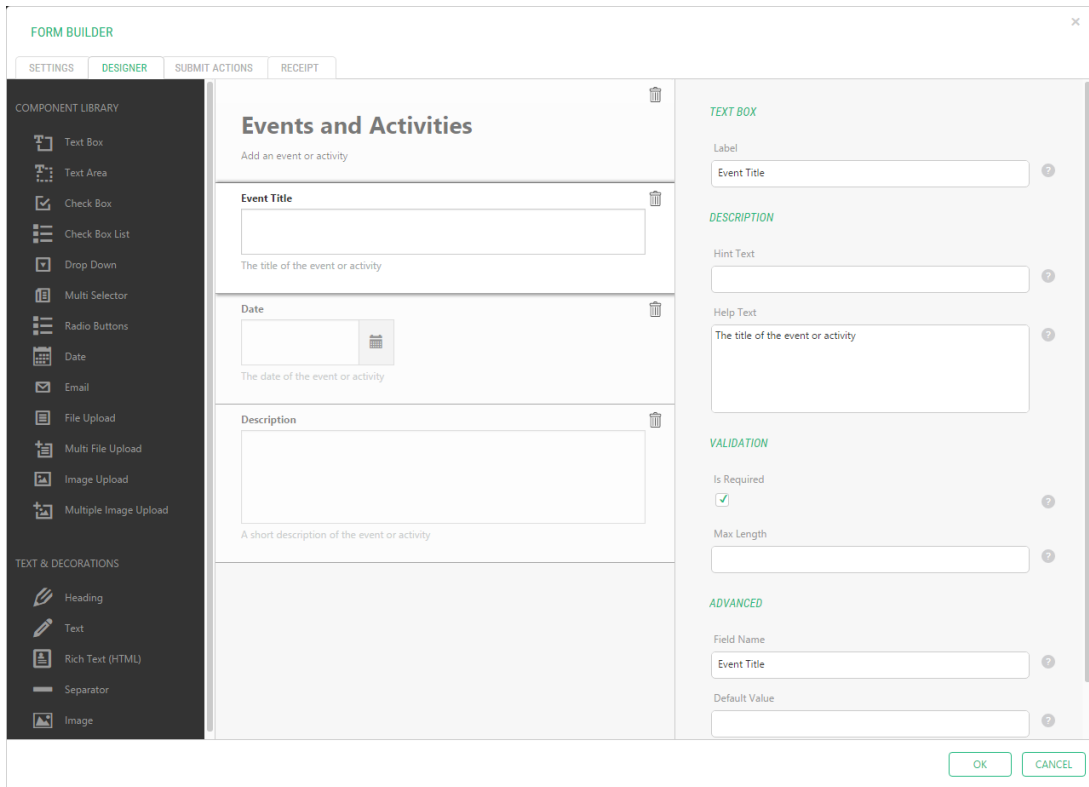
Form Builder - User Guide

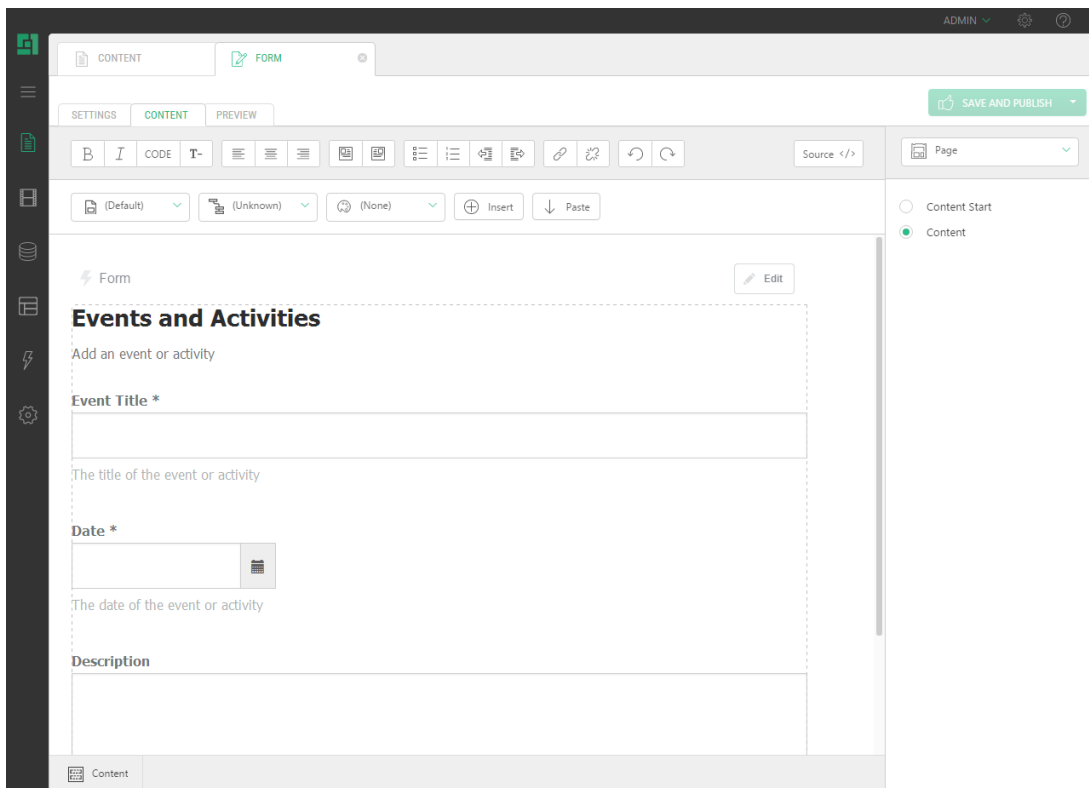Figure 6: Form's visual designer

5. Click **OK**.

Figure 7: The form inserted on a page in Visual Editor

Note. For information about the **Submit Actions** and **Receipt** tabs, please see the corresponding sections further in this guide.

3 Editing Forms

Each form you add on page has a name, fields and a number of other parameters such as the use of CAPTCHA, submit actions or the receipt text.
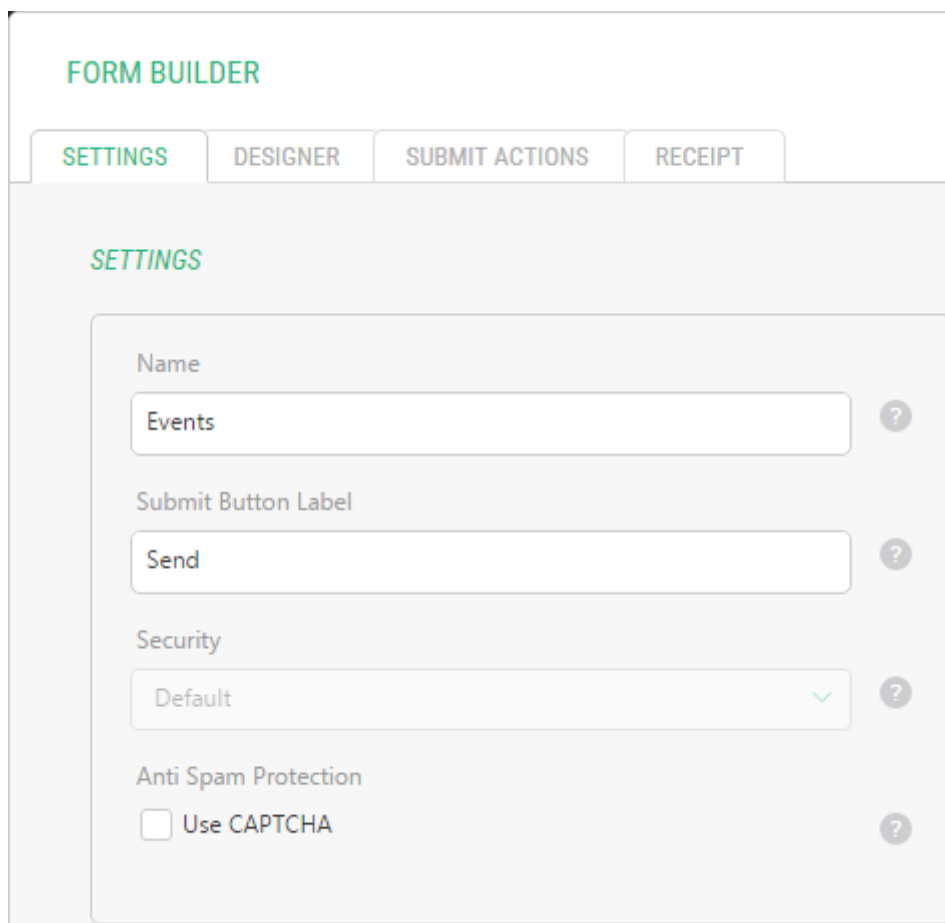
All these components and parameters are conveniently grouped on 4 tabs in the Form Builder window:

- Settings
- Designer
- Submit Actions
- Receipt

3.1 Settings

On the Settings tab, you can specify the form's general settings such as:

- **Name**: The name of the form (mandatory). It identifies the form, for example, helps identify submitted data in Forms Submissions.
- **Submit Button Label**: The label text of the submit button. "Send" by default.
- **Security**: Disabled by default. Allows you to configure your form to use HTTPS
- **Anti Spam Protection**: If the "Use CAPTCHA" option is selected, the CAPTCHA image will be displayed on the form to prevent spam submissions.

Figure 8: Settings tab in the Form Builder

3.1.1 Important Notes

1. If you have two or more forms with the same name, the submitted data from all the forms will be combined.
2. If you rename the form, the data submitted after that will be saved and treated separately according to the form's new name.
3. The security options are disabled by default. Only the default (HTTP) option is available and used. The secure HTTP option is "hidden" in the Form Builder. It can be enabled in the Form Builder's configuration file. (This topic is discussed in the "*Form Builder Developer Guide*".)

3.2 Designer

The Designer tab features a user-friendly WYSIWYG form field editor.

Here, you can conveniently add fields and other components by dragging them from the left pane (Component Library, Text & Decorations) and dropping them on the form (middle pane). Here, you can also change the properties of each field or component on the form properties, for example, its name, label, or default value.
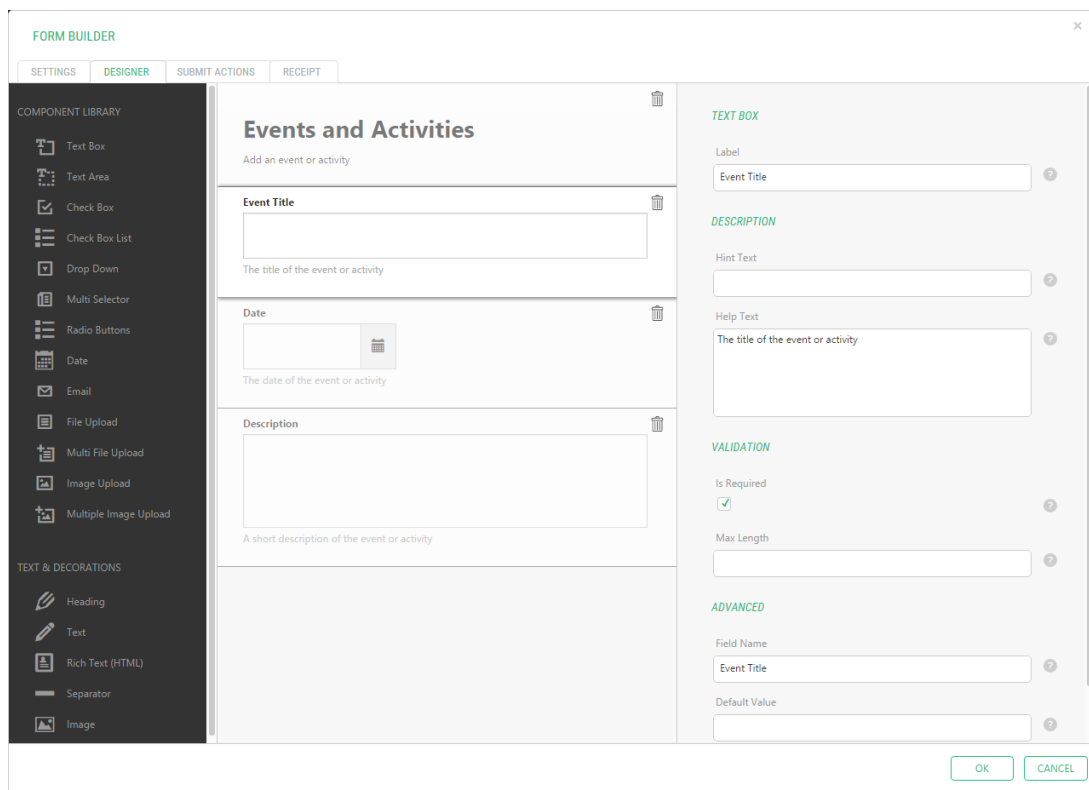

Figure 9: Designer tab in the Form Builder

Please see the corresponding chapters further in this guide about adding fields and editing their properties.

3.3 Submit Actions

On the Submit Actions tab, you can select what actions the form will do when the user submits data.
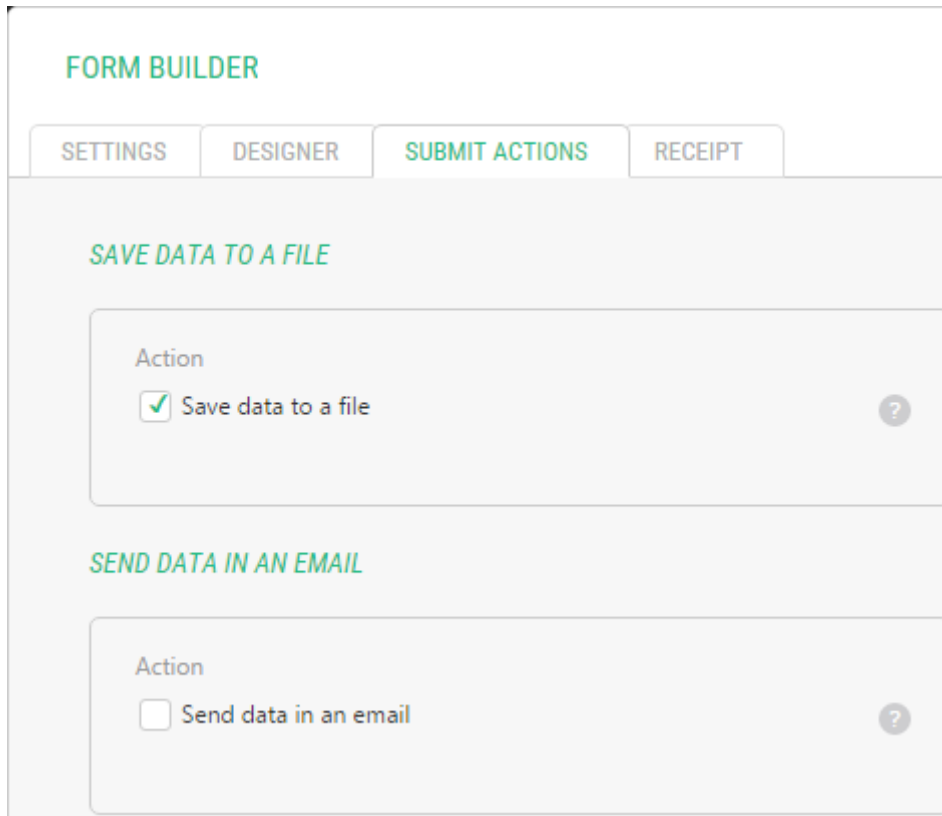
Form Builder - User Guide

Figure 10: Submit Actions tab in the Form Builder

There are two submit actions available out-of-the-box:

- Save to data to a file
- Send data in an email

By default, the option "Save to data to a file" is only selected. All the submitted data are saved to XML files named after the form's name in ~/App_Data/ FormBuilderData on the website.

When selected, the "Send data in an email" option allows you to receive submitted data by email. Please see "Receiving Submitted Data by Email" for more detailed information.

Note. The Form Builder can be extended with custom submit actions and the existing submit actions can be modified or even removed. Custom submit actions are not discussed in this guide and are the topic of the "*Form Builder Developer Guide*".

3.4   Receipt

On the Receipt tab, you can change the text displayed to the user on the page after he or she successfully submits data via the form.
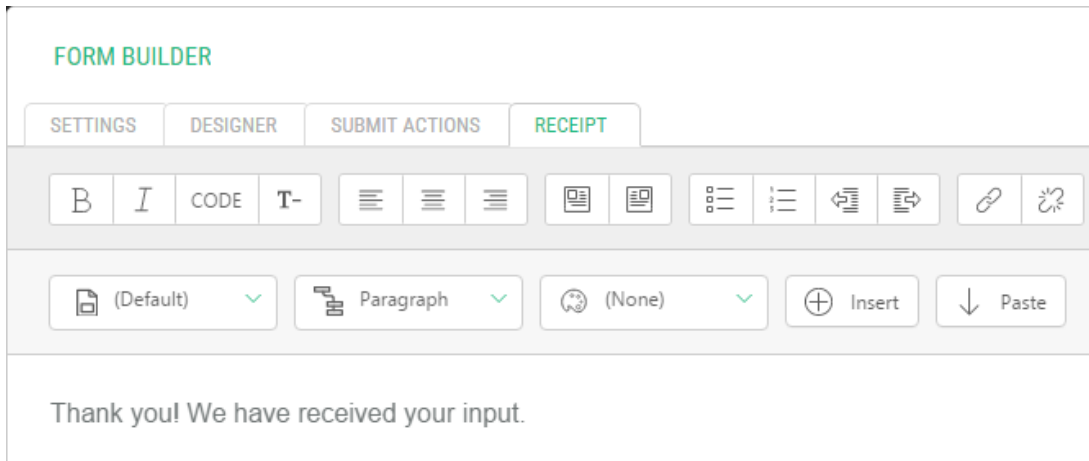
C1 CMS

Figure 11: Receipt tab in the Form Builder

4 Adding Fields to Forms

You can add, edit and remove fields on a form in the Form Builder on the "Designer" tab.

You can simply drag and drop a field component on the form as well as rearrange the fields added already on the form.

When you select a field already added on the form, you get access to its properties where you can fine tune its appearance and behavior.
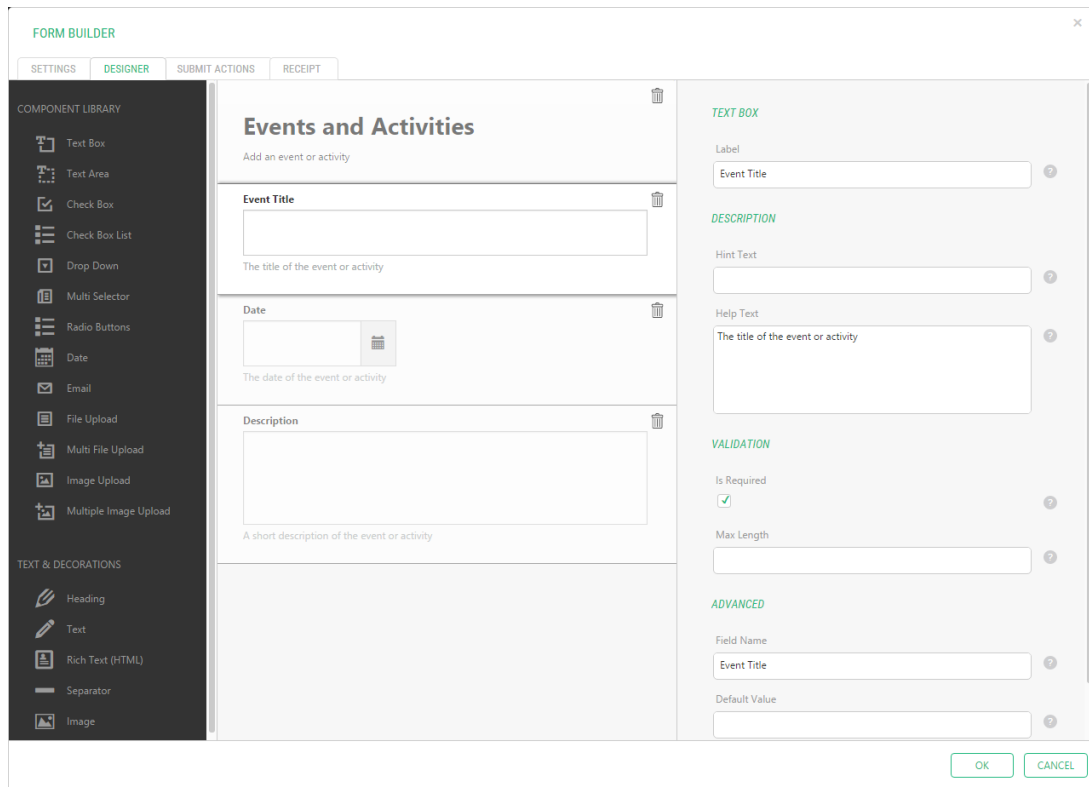

Figure 12: Adding fields to a form in the Form Builder

To add a field to the form:

1. Edit the page where the form is.
2. Edit the properties of the Composite.Forms.FormBuilder.Form function that stands for the form.
3. On the "Designer" tab, from the "Component Library" pane, select, drag and drop a component (field) onto the middle pane. The field will appear in the middle pane, and its properties will be available on the right pane.
4. Where necessary, specify the field's properties such as the name, default value etc. Please note that each component has its own set of properties. (Please see "Editing Field Properties" for more detailed information.)
5. Repeat Steps 3-4 for as many fields as you need.

Note. Along with the field components, you can also use text and decoration components such as headings or separators. They are listed on the right pane under "Text & Decorations".

4.1 Component Library

The components listed in the Component Library are fields of various types you can add to your form. They collect the user input by allowing users to type in, or select values on the form.
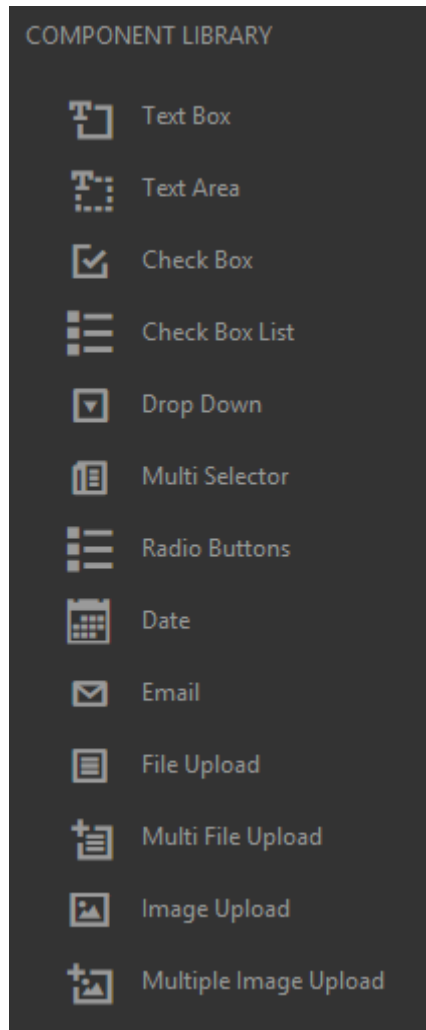
Figure 13: Component Library

Here is a list of the components available out-of-the-box in the Form Builder:

- Text Box
- Text Area
- Check Box
- Check Box List
- Drop Down
- Multi Selector
- Radio Buttons
- Date
- Email
- File Upload
- Multiple Upload
- Image Upload
- Multiple Image Upload

4.2 Text and Decorations

The text and decoration components allows you to improve the form's appearance by headings, texts, images or separators.
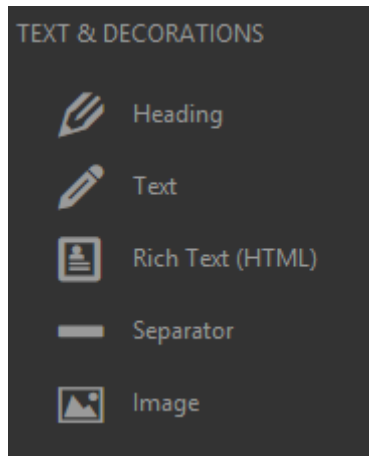
Figure 14: Text and Decorations

Unlike the field components, these components are read-only, they only display information and are not designed for collecting input.

These are the text and decoration components available in the Form Builder out-of-the-box:

- Heading
- Text
- Rich Text (HTML)
- Separator
- Image

C1 CMS

5   Editing Field Properties

You can fine-tune the field added to the form by editing its properties.

Figure 15: Text Box field properties

Most fields that allow users to input values come a number of properties such as:

- Label

- Field Name
- Help Text (the description that normally appears below the field)
- Required (the indication that the value in this field is mandatory; otherwise, the form will not be submitted)

Some fields also allows you to specify:

- the default value
- the hint text (the text that will appear within a field as a hint for an expected value)
- the maximum length

Still some fields might have specific properties. For example, list-like fields will allow you to specify list items (options), media-related fields – paths to the media files.

At minimum an input field will have the Label property which will be filled automatically when added to the form. The Field Name property will be filled automatically after the value in the Label property.

(In general, the scope of properties is defined in the corresponding Razor functions, discussed in depth in the Form Builder Developer Guide.)

Let's have a quick overview of the most common field properties from the out-of-the-box Component Library.

To edit properties of a field:

1. In the Form Builder, select the field the properties of which you want to edit.
2. In the right pane, specify values you need in corresponding properties.
3. Click OK to save the changes to the form fields.

5.1 Label

The Label property identifies the field in the user-friendly manner and appear on the form usually above the field itself.

Last Name *

```
e.g. Smith
```

Enter your last or family name

Figure 16: Label of the field on a page

It is also used identify values submitted by users in report-like views, for example, when you are browsing the submitted data.

Name	Last Name
John	Smith
Jane	Doe

Figure 17: Labels in the submitted data view in the CMS Console

The value of the label property is used to automatically generate the value for the Field Name property.
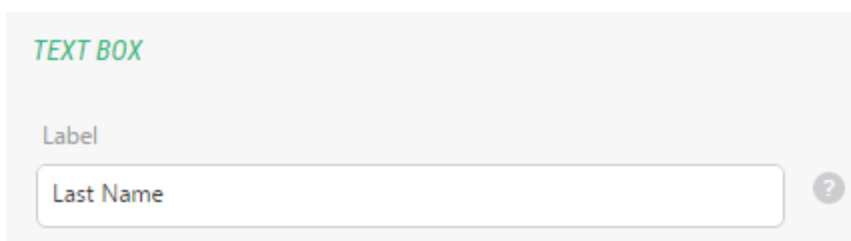
C1 CMS

When you add a field to the form in the Form Builder, the Label property value is generated automatically after the name of the field itself. For example, if you add a Text Box filed, the automatically generated value for the Label property will be "Text Box".

TEXT BOX

Label

Text Box

Figure 18: Default value in the Label property

But you can always replace it with your own value.
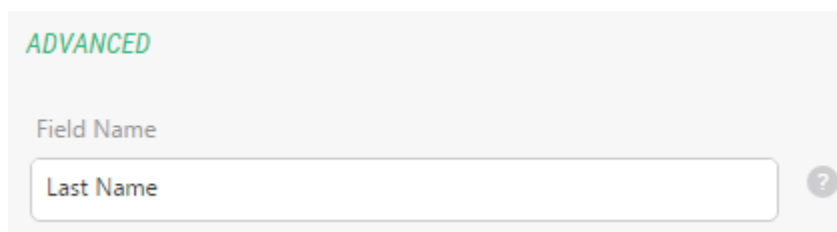
TEXT BOX

Label

Last Name

Figure 19: User-defined value in the Label property

5.2   Field Name

The Field Name property is the programmatic name of the field on the form. It differs from the Label property in that it is used internally by the system, for example, when saving or retrieving data.
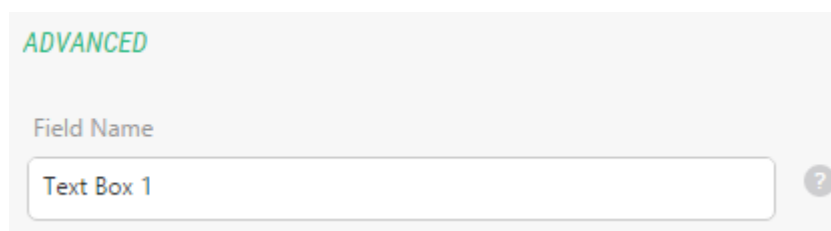
The Field Name property value must be unique for each field on the form.

ADVANCED

Field Name

Last Name

Figure 20: Field Name field property

If not specified explicitly, its value will take after the Label's value. If you happen to have Label properties with the same value on the same form, the automatically generated Field Name's value will get an increasing number appended to it, for example "Text Box 1" for the second Text Box field on the form.

ADVANCED

Field Name

Text Box 1

Figure 21: Auto-generated value for the Field Name property

C1 CMS

You can always replace the automatically generated value with your own provided that it is unique on this form.

5.3   Help Text

The Help Text property contains the description that helps users understand how to fill in the field.
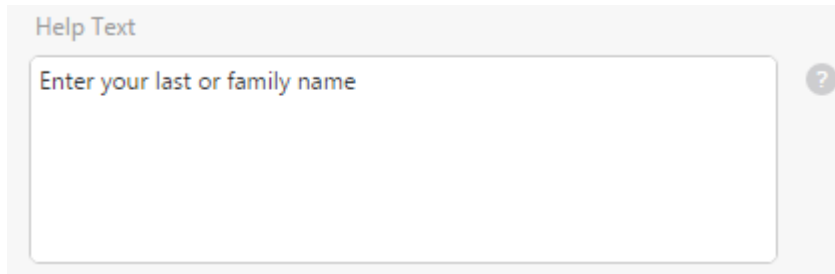

Figure 22: Help Text field property

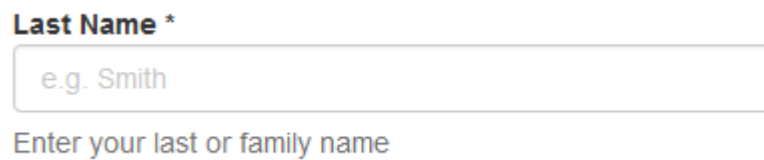The help text normally appears below the field on the form.

Figure 23: Help text below the field on a page

5.4   Required

The Required property allows you to force validation on the field so that it always has a value.
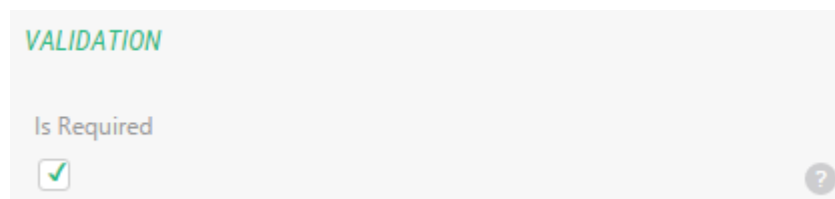

Figure 24: Making a field required

If the field is left blank, the form submission will fail and it will be marked as invalid, for example, by changing the color.

Figure 25: The field has failed validation because it requires input
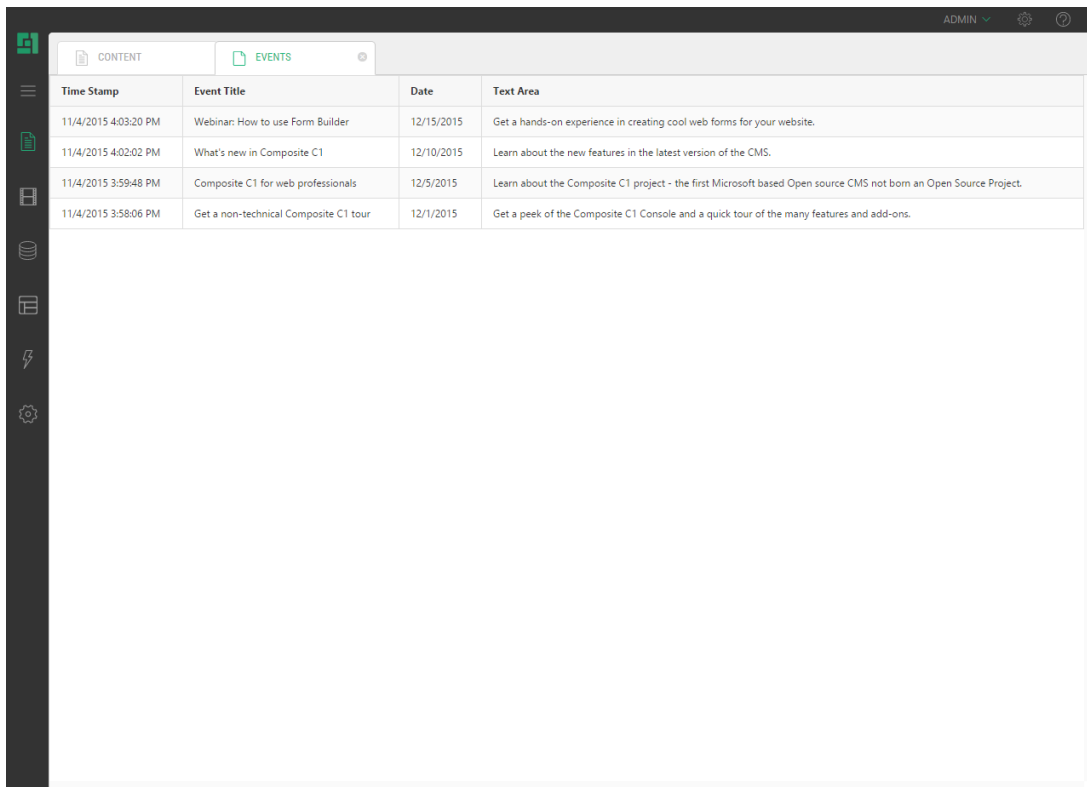
6   Viewing submitted data

You can view submitted data via "Form Submissions". Besides, you can download the submitted data as an XML file or export it to an Excel spreadsheet.

1. In the Content perspective, expand Form Submissions.
2. Select the form the data of which you want to access.
3. Click one the buttons on the button bar:
   - Browse
   - Download as XML
   - Export to Excel

6.1   Browsing Data

To view submitted data in the CMS Console, click the "Browse" button on the toolbar.

The submitted data for the selected form will be displayed in the right pane in table form.

Figure 26: Viewing submitted data in the CMS Console

Physically, the data presented here is stored at ~/App_Data/FormBuilderData/. The files uploaded via the upload-related fields are stored in the subfolders named after the forms.

6.2   Downloading data as XML

To download and view the submitted data as an XML file, click "Download as XML" button.

When prompted, select whether you want to save the XML file or open it in the default XML viewer.

```xml
▼<Form name="Events">
  ▼<FormData timestamp="2013-11-19T14:17:25.314691+02:00" formid="7e06123b-77a2-4b25-8488-1888d42ce55e">
    <Field name="Event Title" value="Get a non-technical Composite C1 tour" type="System.String"/>
    <Field name="Description" value="Get a peek of the Composite C1 Console and a quick tour of the many
    <Field name="Date" value="2013-12-01T00:00:00" type="System.DateTime"/>
  </FormData>
  ▼<FormData timestamp="2013-11-19T14:18:52.2407622+02:00" formid="48762efa-4ccb-4e96-9316-eeb735fef820">
    <Field name="Event Title" value="Composite C1 for web professionals" type="System.String"/>
    <Field name="Description" value="Learn about the Composite C1 project - an Open source CMS not born a
    <Field name="Date" value="2013-12-05T00:00:00" type="System.DateTime"/>
  </FormData>
  ▼<FormData timestamp="2013-11-19T14:22:08.5927958+02:00" formid="cbc34f2f-ecbb-4d0e-b543-a69882373340">
    <Field name="Event Title" value="What's new in Composite C1 4.1?" type="System.String"/>
    <Field name="Description" value="Learn about new features of the latest version of this CMS." type="S
    <Field name="Date" value="2013-12-10T00:00:00" type="System.DateTime"/>
  </FormData>
  ▼<FormData timestamp="2013-11-19T14:23:21.0827868+02:00" formid="12ee7c48-b4c3-4f66-b35c-96485523cb7d">
    <Field name="Event Title" value="Webinar: How to use Form Builder" type="System.String"/>
    <Field name="Description" value="Get a hands-on experience in creating cool web forms for your websit
    <Field name="Date" value="2013-12-15T00:00:00" type="System.DateTime"/>
  </FormData>
</Form>
```

Figure 27: Viewing submitted data in a downloaded XML file

6.3 Exporting to Excel

You can also export the data to Microsoft Excel format and open the data as an Excel spreadsheet by clicking the "Export to Excel" button.

When prompted, select whether you want to save the Excel file or open it in Excel.

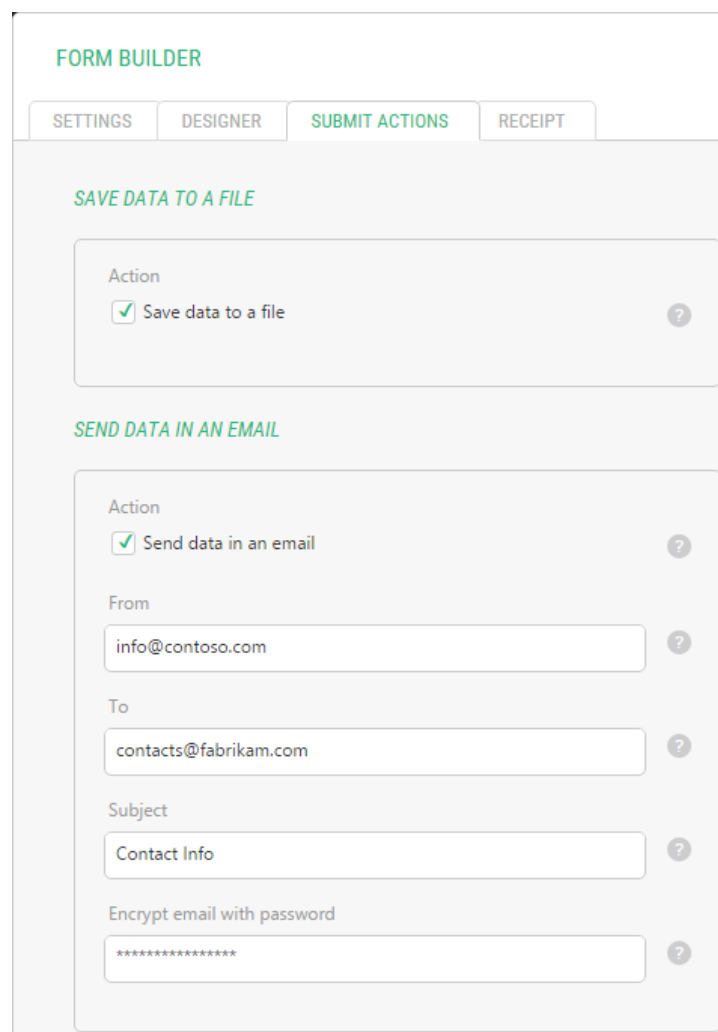Figure 28: Viewing submitted data exported to Excel

7 Receiving Submitted Data by Email

In addition to saving data to a file, you can also configure the form to send the submitted data by email.

Before you set up data submission by email, make sure you've configured the mail host on your website. For more information, please see "Email Host Setup".

To configure data submission by email:

1. Edit the page with the form and edit the Form function properties to open the Form Builder.
2. On the "Submit Actions" tab, select the option "Send data in an email". More fields will become available.
3. Fill out the fields:
    a. **From**: the sender's email address
    b. **To**: the recipient's email address
    c. **Subject**: [optional] the subject of the email. If not specified, the form's title will be used instead.
    d. **Encrypt email with password**: [optional] the password to use to unpack the encrypted ZIP file with submitted data (see "Encrypting submitted data"). If not specified, the submitted data will be sent unencrypted in the message body.
4. Click "OK".

Figure 29: Setting up data submission by email

When the user submits the form, you'll get the email with the data at the address specified in the "To" field above.

7.1   Encrypting submitted data

If you keep the **Encrypt email with password** field blank, you will receive the submitted data by email as a list of filed values in the message body and attachments for all the files uploaded via the form.

If you need your submitted data encrypted, specify the password in the **Encrypt email with password** field. In this case, the submitted data will be saved in an HTML file and both this HTML file and uploaded files if any will be encrypted in a ZIP file.

You will receive this encrypted zip file as an attachment. To view the thus submitted data and files, you'll have to enter the password to unzip the attached file.

8 Test Your Knowledge

TASK 1

1. Edit a page and add a form with the Composite.Forms.FormBuilder.Form function.
2. Add a Text Box field and name it "Name".
3. Add an Email field and name it "Email Address".
4. Name the form "Contacts".
5. Save the form and publish the page.

TASK 2

1. Edit the "Contacts" form created in Task 1.
2. Add the Heading to the top of the form.
3. Provide some arbitrary text for the heading and subheading.
4. Save the form and publish the page.

TASK 3

1. Edit the "Contacts" form created in Task 1.
2. Make both fields – "Name" and "Email Address" – required.
3. Limit the length of the "Name" field to 16 characters.
4. Provide the hint and help texts for both fields.
5. Save the form and publish the page.

TASK 4

1. Edit the "Contacts" form created in Task 1.
2. Enable CAPTCHA on the form.
3. Change the receipt text.
4. Save the form and publish the page.

TASK 5

1. Set up form submission by email on the form.
2. Make sure the emails are sent on the form submission.

TASK 6

1. Open the page with the form in a browser.
2. Submit data of a few contacts via the form.

TASK 7

1. View the submitted data in the CMS Console.
2. Download the submitted data as an XML file.
3. Export the submitted data to Excel.

C1 CMS