

Using Confluence Excel macro with pages created by ConfiForms app



In this tutorial you will learn how to

- how to create a form which will take excel file as an input and stores it as an attachment in Confluence
- how to create pages with ConfiForms IFTTT macro which will show the uploaded file using built in excel macro automatically

NB, You will need <https://marketplace.atlassian.com/plugins/com.atlassian.confluence.plugins.editor.confluence-source-editor> plugin (free) to complete the tutorial. As you have to set the excel macro parameters using the Storage Format editor, because setting dynamic variables via macro dialog is not supported.



If you are new to ConfiForms, please take a few minutes to read the [Basic concepts](#) Guide, or go through the video tutorial on that page. 😊

Since version 1.9.20 of ConfiForms we have added support for using built-in Excel macro (and other Office documents related macros) inside ConfiForms IFTTT.

Here is how you can built a solution which accepts uploading excel files and storing them as attachments and creating a page per file to show the uploaded excel file contents.

Let's start with the ConfiForms Form definition.

We will have only one field in the form and it will be of type "File"

Edit 'ConfiForms Field Definition' Macro

Field for ConfiForms dynamic forms
[Documentation](#)

Field name *

Must be unique within the form defined and alphanumeric

Field label *

☐ Required *

Field type

File - upload files/images

Field description (optional)

Preview

Choose Files No file chosen

[Select macro](#)

SaveCancel

And the form will have an IFTTT macro rule to create a Confluence page when the record is created

The configuration for the form will look like

CONF: ConfiForms Form | formName = f

CONF: ConfiForms Registration Control

CONF: ConfiForms Field Definition | fieldName = excel | fieldLa...

CONF: ConfiForms IFTTT Integration Rules | event = onCreated | action = Create Page | title = page with excel [entry.dateCreatedFo...

Office Excel | page = Excel viewer page generator | name = [



So, you can see we have a ConfiForms form, one registration control to enable creating new records, and a field named "excel" of type "file". Also the form has an IFTTT macro inside which creates a page when new record is saved and shows the file uploaded using the excel macro. This is how the form looks like in view mode.

Excel viewer page generator

Created by admin, last modified just a moment ago

Register Form: f

Registration

Excel file

Choose Files No file chosen

+

Save Close

13:22

15:24

The interesting and most important part is how the excel macro is configured.



Let's see its contents using the storage format viewer (You will need a <https://marketplace.atlassian.com/plugins/com.atlassian.confluence.plugins.editor.confluence-source-editor> plugin to complete the tutorial)

Source Editor

Find Replace Previous Next Replace Replace all ☐ Regex ☐ Match case

```

1 <ac:structured-macro ac:macro-id="6b03b5c6-e20e-46ef-86f3-9fe2989a84b6" ac:name="conform" ac:schema-version="1">
2   <ac:parameter ac:name="formName">f</ac:parameter>
3   <ac:rich-text-body>
4     <p>
5       <ac:structured-macro ac:macro-id="ef231423-e68d-4dea-b3e7-d4f1f91254d7" ac:name="conform-entry-register" ac:schema-version="1">
6         <ac:parameter ac:name="atlassian-macro-output-type">INLINE</ac:parameter>
7         <ac:rich-text-body>
8           <p> </p>
9         </ac:rich-text-body>
10      </ac:structured-macro>
11      <ac:structured-macro ac:macro-id="476fadb2-1b40-4e79-8a43-fc76634922ab" ac:name="conform-field-definition" ac:schema-version="1">
12        <ac:parameter ac:name="fieldName">excel</ac:parameter>
13        <ac:parameter ac:name="fieldLabel">Excel file</ac:parameter>
14        <ac:parameter ac:name="type">file</ac:parameter>
15      </ac:structured-macro>
16    </p>
17    <ac:structured-macro ac:macro-id="e63be7d7-6d67-4ea3-b122-935636026377" ac:name="conform-ifttt" ac:schema-version="1">
18      <ac:parameter ac:name="action">Create Page</ac:parameter>
19      <ac:parameter ac:name="event">onCreated</ac:parameter>
20      <ac:parameter ac:name="title">page with excel [entry.dateCreatedFormatted]</ac:parameter>
21      <ac:rich-text-body>
22        <p>
23          <ac:structured-macro ac:macro-id="f92dee96-d52a-46c2-88a1-ab5a327cfff1b" ac:name="viewxls" ac:schema-version="1">
24            <ac:parameter ac:name="name">
25              <ri:attachment ri:filename="[entry.excel]" />
26            </ac:parameter>
27            <ac:parameter ac:name="page">
28              <ac:link>
29                </ac:parameter>
30              <ac:parameter ac:name="space">
31                <ri:space ri:space-key="test" />
32              </ac:parameter>
33            </ac:structured-macro>
34          </p>
35        </ac:rich-text-body>
36      </ac:structured-macro>
37    </ac:rich-text-body>
38  </ac:structured-macro>

```

☐ Wrap long lines Apply Cancel

The important part here is highlighted, you can see that "filename" is set to [entry.excel], which will be dynamically substituted to the uploaded file name when the page is created.

You can see that the page parameter is set to point at current page, **this is important**. As this link will be correctly substituted to point at original page (at current page), not at the page created

And we have also included the space parameter to set the Confluence space explicitly (we have our page in TEST space)



In some case you will need to update the "view-file" macro's "page" parameter to be

```

<ac:parameter ac:name="page">
  <ac:link>
    <ri:page ri:content-title="[entry._page.title]" />
  </ac:link>
</ac:parameter>

```

and remove the "space" parameter (if the page is created in the same Confluence space)

And the result is as follows (we have set a naming convention for created pages to "page with excel [entry.dateCreatedFormatted]", as you may see in the storage format above)

page with excel 2015-09-16 15:41

Created by admin 15 minutes ago

[Edit Document](#)

Id	Created By	Owned By	Created	Created By	Owned By	User
1670e22b-2eb1-4772-9540-e46c136550e6	admin	admin	1429124978414	admin	admin	Sasha
05d239ee-5c50-44e0-b846-0b0ed0308b4a	admin	admin	1429125083209	admin	admin	Sasha
d823553d-bd36-4b20-	admin	admin	1429125174571	admin	admin	Sasha