

Sending email from Confluence with ConfiForms IFTTT macro conditionally

 In this tutorial you will learn how to send conditional emails with ConfiForms IFTTT macro, and also some tricks around it.

This tutorial uses the concept of IFTTT results chaining, described here: [Advanced integrations with IFTTT macros with ConfiForms](#)

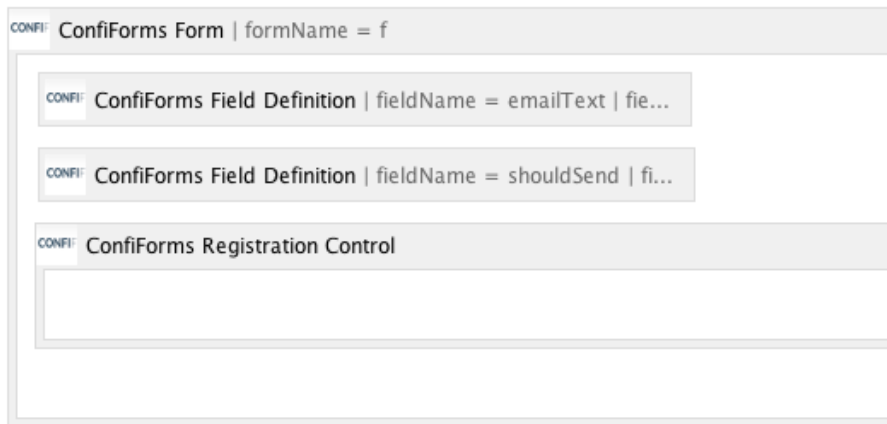
 If you are new to ConfiForms, please take a few minutes to read the [Basic concepts](#) Guide, or go through the video tutorial on that page. 😊

Quite often we have been asked if ConfiForms could send emails with form data but only "when some checkbox is checked". And we have a solution for you to share. It is a bit advanced, so if you are very new to ConfiForms, please read the tutorial about basics and then come back. Also a concept about IFTTT results chaining and how to reference and use it would be beneficial to learn (See link above).

We will start, as usual, with form definition.

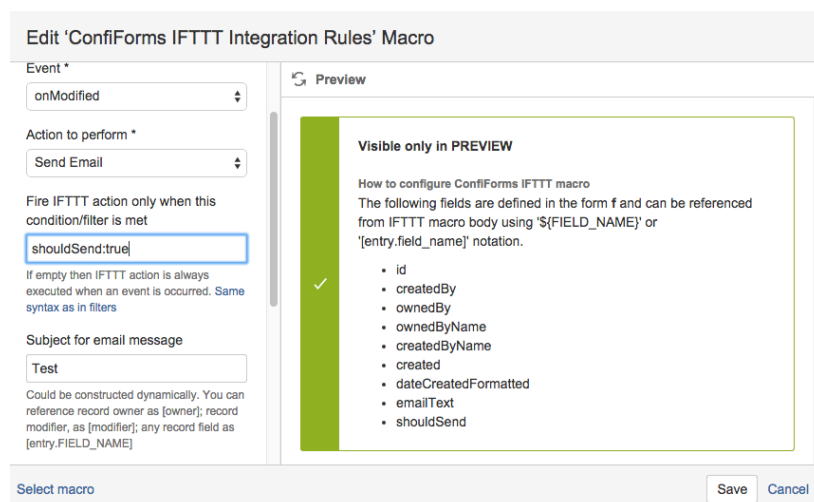
Simple form with just 2 fields: One of type "text" and another is of type "checkbox"

"Checkbox" field will tell us when an email should be send out.



The screenshot shows the 'ConfiForms Form' configuration interface. At the top, it says 'ConfiForms Form | formName = f'. Below this, there are two 'ConfiForms Field Definition' blocks. The first block is for 'emailText' and the second is for 'shouldSend'. Below these fields, there is a 'ConfiForms Registration Control' section which is currently empty.

Then we add 2 IFTTT handlers (we set to catch only onModified events, so if you want to have the same logic for new items you will need to add 2 more similar handlers, which will be executed on "onCreated" event). For simplicity we define only 2 here, one for sending emails when checkbox is selected and another one to put the checkbox field value back to "false". See screenshots below



The screenshot shows the 'Edit 'ConfiForms IFTTT Integration Rules' Macro' configuration window. On the left, under 'Event', 'onModified' is selected. Under 'Action to perform', 'Send Email' is selected. The 'Fire IFTTT action only when this condition/filter is met' section has 'shouldSend:true' entered. Below this, there is a note: 'If empty then IFTTT action is always executed when an event is occurred. Same syntax as in filters'. The 'Subject for email message' is set to 'Test'. On the right, a 'Preview' section shows a list of fields defined in the form 'f': id, createdBy, ownedBy, ownedByName, createdByName, created, dateCreatedFormatted, emailText, and shouldSend. A green checkmark is next to the list. At the bottom, there are 'Save' and 'Cancel' buttons.

See the "condition/filter" we set here: IFTTT is executed only if the checkbox value is set

Edit 'ConfiForms IFTTT Integration Rules' Macro

Action to perform:

Create / Update ConfiForms Ent

Fire IFTTT action only when this condition/filter is met

If empty then IFTTT action is always executed when an event is occurred. Same syntax as in filters

Parameters to set on the entry

entryId=[entry.id]&sendEmail=

As key-value pairs, separated by &. Where key is the field name in the form where this entry should be created and value is the value to be set. Could be constructed dynamically. You can reference record owner as [owner]; record modifier, as [modifier]; any record field as [entry.FIELD_NAME]

Preview

Visible only in PREVIEW

How to configure ConfiForms IFTTT macro

The following fields are defined in the form f and can be referenced from IFTTT macro body using '{FIELD_NAME}' or '{entry.field_name}' notation.

- id
- createdBy
- ownedBy
- ownedByName
- createdByName
- created
- dateCreatedFormatted
- emailText
- shouldSend

Select macro Save Cancel

Here we update the ConfiForms entry "back" after the email is sent. We can actually "optimize" it here a bit and only execute when checkbox is actually checked, but for simplicity we choose to execute it each time, and reset the value for checkbox on each "save".

This configuration allows us to support cases when email should be sent only when some field has a certain value set and then we are able to reset the field back to original/default value automatically.

Here is how the form configuration looks in Confluence editor, the ordering of IFTTT handlers is important

ConfiForms Form | formName = f

ConfiForms Field Definition | fieldName = emailText | fie...

ConfiForms Field Definition | fieldName = shouldSend | fl...

ConfiForms Registration Control

ConfiForms IFTTT Integration Rules | event = onModified | action = Send Email | condition = shouldSend:true | title = Test [...]

ConfiForms IFTTT Integration Rules | event = onModified | action = Create ConfiForms Entry | title = entryId=[entry.id]&send...

Edit

Remove

And the final view of the form when user enters data:

Registration

Email text

test

Send email

☒

Send update via email

+

Save

Close

