

Creating quick feedback forms with ConfiForms



This tutorial demonstrates advanced capabilities of ConfiForms app and it's APIs and shows you how to build a "non boring" forms for collecting user's feedback

We at [Vertuna LLC](#), live, eat and breath data entry forms, we just love to build forms to quickly collect user opinions and feedback. And we love Atlassian products, so we have built, and continue to build, the best forms app for Confluence available on Atlassian Marketplace called - [ConfiForms](#), [Data entry forms and workflows](#).

ConfiForms provides a **lot of versatility on the features and use cases it covers and supports**. While, almost everything could be build with Confluence editor, using the macros ConfiForms provides, in this article we would like to show you some custom, sexy forms, you can build with ConfiForms and with a bit of scripting.

We will build a form to quickly gather user's feedback, something you have probably seen in the airports all over the globe

[blocked URL](#)

An example of how ongoing feedback is collected at Heathrow Airport [\[Source\]](#)

Looks familiar? 😊

OK, in our case it will look a little bit different and will be something like this



Icons designed by from [Flaticon](#) [\[License\]](#)

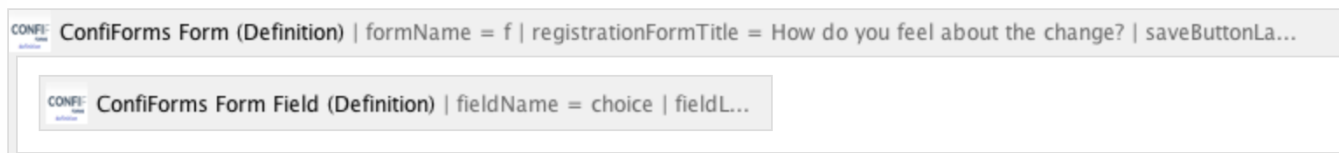
We would like to collect the data from our users and show some simple charts and reports over the collected data later.

Let's get this going.

We will use ConfiForms plugin to define the form, The form will be very simple, having just one field definition and the field definition would be of type "Dropdown" with 4 options

id	label
1	Awesome
2	Good
3	Could be better
4	WTF!?

In Confluence editor this would look like this:



And the *choice* field configured as something like this

Edit 'ConfiForms Form Field (Definition)' Macro

Documentation

Field name *

Must be unique within the form defined and can be alphanumeric

Field label *

☒ Required *

Field type

id	label
1	Awesome <input type="text"/> <input type="button" value="✖"/>
2	Good <input type="text"/> <input type="button" value="✖"/>

Preview

Select macro

Save

Cancel

We can add a default FormView, but it is rather boring, as you can see below

How do you feel about the change?

Please share

We want something more sexy and user engaging. So, we will build a user macro with icons and a little bit of scripting to glue the things together.



When a form is used by an authenticated user then this information is stored in the *createdBy* field, as ConfiForms has some metadata fields stored with each record (even when you have defined just one field, as we did in this example)

When a form is used by anonymous users the *createdBy* field is left blank

Each image has a custom "onClick" handler and invokes ConfiForms [REST API](#) to create new record in the form

The code, the handler invokes, looks something like this:

```

// mychoice is the value (ID) for each option in "choice" field

function leaveFeedback(mychoice) {
  AJS.$.ajax({
    url: 'https://wiki.vertuna.com/ajax/confiforms/rest/save.action?pageId=16646199&f=f&choice=' + mychoice,
    type: "GET",
    success: function (data) {
      // here we need to tell we received the feedback
    }
  });
}

```

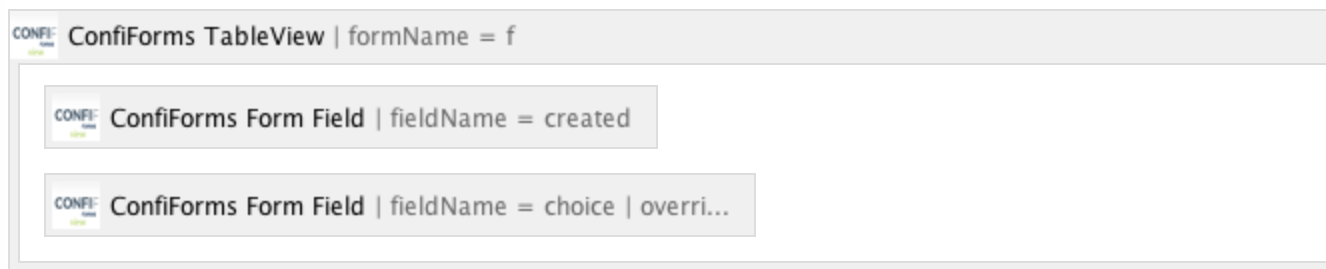
See more complete example on how to setup the handlers with scripting here: [Custom feedback form with ConfiForms and some JavaScript](#)

You can also see the form in action and user macro created for you

- [Creating quick feedback forms with ConfiForms - the form](#)
- [Creating quick feedback forms with ConfiForms - user macro source code](#)

What is good about ConfiForms is that you can use and re-use the defined form on multiple pages, as well as build the "views" around the stored data where you want. Let's build the table over the stored data, we will show the metadata field *created* with the feedback given

This is as simple as this, thanks to ConfiForms TableView macro



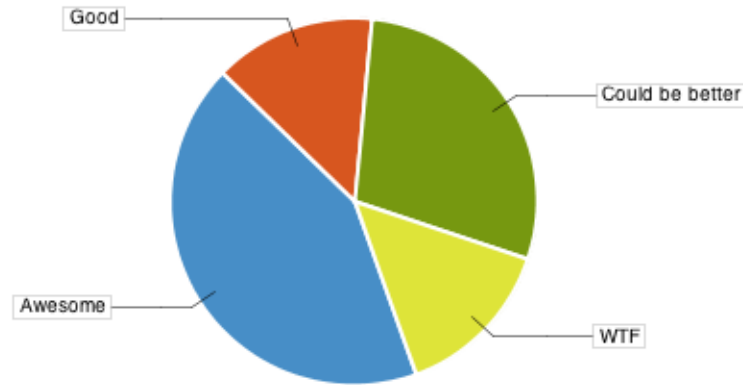
And the result would be something like this

Created	User feedback
Oct 06, 2017 16:06	Awesome
Oct 06, 2017 16:06	Could be better
Oct 06, 2017 16:06	WTF!?
Oct 06, 2017 16:06	Good
Oct 06, 2017 16:06	Awesome
Oct 06, 2017 16:06	Could be better
Oct 06, 2017 16:06	Awesome

Of course we can add some styling to the table, and can show other fields stored (like the mentioned earlier *createdBy* metadata field) or with modifying the form definition a little bit further we can collect some system environment data, see notes on the bottom of this page.

Also, to have a good overview of the data stored we will make a simple report using Confluence built in chart macro and will group the choices made by our users and present the aggregated data as a pie chart

Awesome	Good	Could be better	WTF
3	1	2	1



■ Awesome (3 - 43%) ■ Good (1 - 14%) ■ Could be better (2 - 29%) ■ WTF (1 - 14%)

All in real time, rendering the chart based on the data stored, by using ConfiForms Table View Merger macro together with Confluence Chart macro

Chart | orientation = horizontal | width = 700 | dataOrientation = horizontal

ConfiForms TableView Merger

- ConfiForms Form Field | fieldName = No choice:([cou...
- ConfiForms Form Field | fieldName = Awesome:([count...
- ConfiForms Form Field | fieldName = Good:([count_ch...
- ConfiForms Form Field | fieldName = Could be better...
- ConfiForms Form Field | fieldName = WTF:([count_cho...

ConfiForms TableView | formName = f

We can add some invisible to user feature that will helps us to understand more about the users and where the feedback has been actually given.

- We can add a ConfiForms Field Definition of type "IP" to track the IP addresses of the users (if that is what you want)
- We can add a ConfiForms Field Definition of type "Autopage" to track automatically where the feedback has been given. Remember that you can define the form once and then use it anywhere in your Confluence, on any page / blogpost. So, it might be interesting to know where the feedback was actually given

All these fields are not visible to end-users but enable better visibility over the data we collect.