

Rendering Options

Where the options of an element come from, which one decides, and the options of the layouts, the groups and the columns

Anonymous · 15/10/2026

Rendering Options

The way an element is rendered depends on its **options**, and they can be given in several places. The one that is closest to the element decides.

Which option decides

From the one that decides first to the one that decides last:

1. **The options of the element:** the `options` of its definition in the UI schema.
2. **The options that are given when rendering:** the second argument of the Twig functions (`form(form, options)`, `form_row(field, options)`, `form_element(element, form, options)`...) or the `$options` of the renderer. They go down to all the elements, so they are the defaults of the whole tree.
3. **The options of the form:** the `options` section of its definition.
4. **The defaults of the renderer.**

```
'options' => ['floating_labels' => false],           // 3. All the fields.
```

```
{{ form(form, {floating_labels: true}) }}           {# 2. Overrides the form. #}
```

```
['type' => 'Control', 'scope' => '#/properties/name', 'options' => ['floating_labels'  
=> false]], // 1. Wins.
```

The options of a layout, of a group or of a categorization are for itself: its children do not get them.

Options of the layouts

Element	Option	What it does
VerticalLayout	<code>layout_attr</code>	Attributes of its container (<code>class</code> , <code>id</code> , <code>data-*</code> ...)
HorizontalLayout	<code>layout_attr</code>	Attributes of its row
Group	<code>group_attr</code>	Attributes of its card

Element	Option	What it does
Group	<code>collapsible</code>	The group can be collapsed
Group	<code>collapsed</code>	It starts collapsed
Group	<code>footer</code>	HTML of the footer of the card
Categorization	<code>categorization_attr</code>	Attributes of its container
Categorization	<code>tab_style</code>	<code>horizontal</code> (by default) or <code>vertical</code>
Categorization	<code>vertical_col_width</code>	Columns of the tabs when they are vertical (3 by default)
Category	<code>category_attr</code>	Attributes of the content of the tab

```
[
  'type' => 'Group',
  'label' => 'Address',
  'options' => ['collapsible' => true, 'collapsed' => true, 'footer' => 'We never
share it.'],
  'elements' => [/* ... */],
],
```

Columns of a horizontal layout

The elements of a `HorizontalLayout` are columns of the row. The ones that do not say their size share what is left, **in equal parts**, so with any number of elements the row is full. An element asks for a size with `col_size`, the number of columns of Bootstrap (from 1 to 12, or auto):

```
[
  'type' => 'HorizontalLayout',
  'elements' => [
    ['type' => 'Control', 'scope' => '#/properties/street', 'options' =>
['col_size' => 8]],
    ['type' => 'Control', 'scope' => '#/properties/number'],    // The rest: 4.
  ],
],
```

The class of the column is `col-md-8` for the first one and `col-md` for the second.

Options for the children of a layout

A layout or a group can give options to each one of its children by its position, with `elements_options`: a list where the entry `n` is for the child `n`. They count as options that are given to that child when rendering, so the options of the child itself decide before them (see above).

```
[
  'type' => 'Group',
  'label' => 'Address',
  'options' => [
    'elements_options' => [
      ['attr' => ['autocomplete' => 'street-address']], // For the first
child.
      [], // The second one has
none.
    ],
  ],
  'elements' => [/* ... */],
],
```

In a `HorizontalLayout` the entry of a child is also the one of its **column**: its `col_size` and its `class` are taken from it, and they decide before the `col_size` that the child asks for.

```
'options' => ['elements_options' => [['col_size' => 3, 'class' => 'text-end'], []]],
```

The end of the form

`form_end` renders the closing tag, and before it the fields that were not rendered yet (`form_rest`), with the same options as the rest of the form. The option `render_rest` set to `false` does not render them:

```
{{ form_start(form) }}
  {{ form_row(form.fields.name) }}
{{ form_end(form, {render_rest: false}) }}
```

Last updated on 15/10/2026