django-bootstrap4 Documentation

Release 0.0.5

Dylan Verheul

Feb 14, 2018

Contents

Contents:

CHAPTER 1

Installation

The preferred way to install `django-bootstrap4` is `pip`:

```
$ pip install django-bootstrap4
```

Alternatively, you can install download or clone this repo and install from its folder with:

```
$ pip install -e .
```

In your project, you should add `django-bootstrap4` to your `requirements.txt`.

Be sure to use `virtualenv` if you develop python projects.

Add to INSTALLED_APPS in your `settings.py`:

```
'bootstrap4',
```

After installation, the *Quickstart* will get you on your way to using `django-bootstrap4`.

CHAPTER 2

Quickstart

After *Installation*, you can use `django-bootstrap4` in your templates.:

Load the `bootstrap4` library and use the `bootstrap_*` tags:

2.1 Example template

```
{# Load the tag library #}
{% load bootstrap4 %}

{# Load CSS and JavaScript #}
{% bootstrap_css %}
{% bootstrap_javascript jquery='full' %}

{# Display django.contrib.messages as Bootstrap alerts #}
{% bootstrap_messages %}

{# Display a form #}
<form action="/url/to/submit/" method="post" class="form">
  {% csrf_token %}
  {% bootstrap_form form %}
  {% buttons %}
    <button type="submit" class="btn btn-primary">
      {% bootstrap_icon "star" %} Submit
    </button>
  {% endbuttons %}
</form>

{# Read the documentation for more information #}
```

2.2 Template tags and filters

Refer to *Template tags and filters* for more information.

2.3 Settings

You can set defaults for `django-bootstrap4` in your settings file. Refer to *Settings* for more information.

2.4 Demo application

The demo application provides a number of useful examples.

https://github.com/dyve/django-bootstrap4/tree/develop/demo

CHAPTER 3

Migration

Below is a list of caveats when migrating from Bootstrap3/django-bootstrap3 to Bootstrap4/django-bootstrap4.

This document only considers the differences between django-bootstrap3 and django-bootstrap4. For the migration guide from Bootstrap 3 to 4, please look at the Bootstrap docs, especially the [Migration section](https://v4-alpha.getbootstrap.com/migration/).

3.1 Icons

Bootstrap 4 does not ship with an icon set, so everything regarding to icons has been removed.

If you need easy access to icons that are compatible with Bootstrap 4, please consider https://github.com/zostera/django-fa.

CHAPTER 4

Template tags and filters

Note: All the following examples it is understood that you have already loaded the `bootstrap4` template tag library, placing the code below in the beginning that each template that `bootstrap4` template tag library will be used. Read the *Installation* and *Quickstart* sections on how to accomplish this.

4.1 bootstrap_form

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_form`**`(`**args*, ***kwargs*`)`

Render a form

Tag name:

```
bootstrap_form
```

Parameters:

form The form that is to be rendered

exclude A list of field names (comma separated) that should not be rendered E.g. exclude=subject,bcc

See *bootstrap_field* for other arguments

Usage:

```
{% bootstrap_form form %}
```

Example:

```
{% bootstrap_form form layout='inline' %}
```

4.2 bootstrap_form_errors

bootstrap4.templatetags.bootstrap4.**bootstrap_form_errors**(**args*, ***kwargs*)

Render form errors

Tag name:

```
bootstrap_form_errors
```

Parameters:

form The form that is to be rendered

type Control which type of errors should be rendered.

One of the following values:

- 'all'
- 'fields'
- 'non_fields'

default 'all'

layout Context value that is available in the template bootstrap4/form_errors.html as layout.

Usage:

```
{% bootstrap_form_errors form %}
```

Example:

```
{% bootstrap_form_errors form layout='inline' %}
```

4.3 bootstrap_formset

bootstrap4.templatetags.bootstrap4.**bootstrap_formset**(**args*, ***kwargs*)

Render a formset

Tag name:

```
bootstrap_formset
```

Parameters:

formset The formset that is being rendered

See *bootstrap_field* for other arguments

Usage:

```
{% bootstrap_formset formset %}
```

Example:

```
{% bootstrap_formset formset layout='horizontal' %}
```

4.4 bootstrap_formset_errors

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_formset_errors`**(**args*, ***kwargs*)

Render formset errors

Tag name:

```
bootstrap_formset_errors
```

Parameters:

formset The formset that is being rendered

layout Context value that is available in the template `bootstrap4/form_errors.html` as `layout`.

Usage:

```
{% bootstrap_formset_errors formset %}
```

Example:

```
{% bootstrap_formset_errors formset layout='inline' %}
```

4.5 bootstrap_field

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_field`**(**args*, ***kwargs*)

Render a field

Tag name:

```
bootstrap_field
```

Parameters:

field The form field to be rendered

layout If set to `'horizontal'` then the field and label will be rendered side-by-side, as long as there is no `field_class` set as well.

form_group_class CSS class of the `div` that wraps the field and label.

default `'form-group'`

field_class CSS class of the `div` that wraps the field.

label_class CSS class of the `label` element. Will always have `control-label` as the last CSS class.

show_help Show the field's help text, if the field has help text.

default `True`

show_label Whether the show the label of the field.

default `True`

exclude A list of field names that should not be rendered

size Controls the size of the rendered `div.form-group` through the use of CSS classes.

One of the following values:

- `'small'`
- `'medium'`
- `'large'`

placeholder Sets the placeholder text of a textbox

horizontal_label_class Class used on the label when the `layout` is set to `horizontal`.

default `'col-md-3'`. Can be changed in *Settings*

horizontal_field_class Class used on the field when the `layout` is set to `horizontal`.

default `'col-md-9'`. Can be changed in *Settings*

addon_before Text that should be prepended to the form field. Can also be an icon, e.g. `'<span class="glyphicon glyphicon-calendar"></span>'`

See the *Bootstrap docs <http://getbootstrap.com/components/#input-groups-basic>* for more examples.

addon_after Text that should be appended to the form field. Can also be an icon, e.g. `'<span class="glyphicon glyphicon-calendar"></span>'`

See the *Bootstrap docs <http://getbootstrap.com/components/#input-groups-basic>* for more examples.

addon_before_class Class used on the span when `addon_before` is used.

One of the following values:

- `'input-group-addon'`
- `'input-group-btn'`

default `input-group-addon`

addon_after_class Class used on the span when `addon_after` is used.

One of the following values:

- `'input-group-addon'`
- `'input-group-btn'`

default `input-group-addon`

error_css_class CSS class used when the field has an error

default `'has-error'`. Can be changed *Settings*

required_css_class CSS class used on the `div.form-group` to indicate a field is required

default `''`. Can be changed *Settings*

bound_css_class CSS class used when the field is bound

default `'has-success'`. Can be changed *Settings*

Usage:

```
{% bootstrap_field field %}
```

Example:

```
{% bootstrap_field field show_label=False %}
```

4.6 bootstrap_label

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_label`**`(`**args*, ***kwargs*`)`

Render a label

Tag name:

```
bootstrap_label
```

Parameters:

content The label's text

label_for The value that will be in the `for` attribute of the rendered `<label>`

label_class The CSS class for the rendered `<label>`

label_title The value that will be in the `title` attribute of the rendered `<label>`

Usage:

```
{% bootstrap_label content %}
```

Example:

```
{% bootstrap_label "Email address" label_for="exampleInputEmail1" %}
```

4.7 bootstrap_button

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_button`**`(`**args*, ***kwargs*`)`

Render a button

Tag name:

```
bootstrap_button
```

Parameters:

content The text to be displayed in the button

button_type Optional field defining what type of button this is.

Accepts one of the following values:

- `'submit'`
- `'reset'`
- `'button'`
- `'link'`

button_class The class of button to use. If none is given, btn-default will be used.

extra_classes Any extra CSS classes that should be added to the button.

size Optional field to control the size of the button.

Accepts one of the following values:

- `'xs'`
- `'sm'`
- `'small'`
- `'md'`
- `'medium'`
- `'lg'`
- `'large'`

href Render the button as an `<a>` element. The `href` attribute is set with this value.

name Value of the `name` attribute of the rendered element.

value Value of the `value` attribute of the rendered element.

Usage:

```
{% bootstrap_button content %}
```

Example:

```
{% bootstrap_button "Save" button_type="submit" button_class="btn-primary" %}
```

4.8 bootstrap_alert

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_alert`**(*content*, *alert_type=u’info’*, *dismissable=True*)

Render an alert

Tag name:

```
bootstrap_alert
```

Parameters:

content HTML content of alert

alert_type

- `'info'`
- `'warning'`
- `'danger'`
- `'success'`

default `'info'`

dismissable boolean, is alert dismissable

default True

Usage:

```
{% bootstrap_alert content %}
```

Example:

```
{% bootstrap_alert "Something went wrong" alert_type='error' %}
```

4.9 buttons

bootstrap4.templatetags.bootstrap4.**bootstrap_buttons**(*parser*, *token*)

Render buttons for form

Tag name:

```
buttons
```

Parameters:

submit Text for a submit button

reset Text for a reset button

Usage:

```
{% buttons %}{% endbuttons %}
```

Example:

```
{% buttons submit='OK' reset="Cancel" %}{% endbuttons %}
```

4.10 bootstrap_messages

bootstrap4.templatetags.bootstrap4.**bootstrap_messages**(*context*, **args*, ***kwargs*)

Show django.contrib.messages Messages in Bootstrap alert containers.

In order to make the alerts dismissable (with the close button), we have to set the jquery parameter too when using the bootstrap_javascript tag.

Uses the template `bootstrap4/messages.html`.

Tag name:

```
bootstrap_messages
```

Parameters:

None.

Usage:

```
{% bootstrap_messages %}
```

Example:

```
{% bootstrap_javascript jquery=True %}
{% bootstrap_messages %}
```

4.11 bootstrap_pagination

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_pagination`**`(`*page*, ***kwargs*`)`

Render pagination for a page

Tag name:

```
bootstrap_pagination
```

Parameters:

page The page of results to show.

pages_to_show Number of pages in total

default `11`

url URL to navigate to for pagination forward and pagination back.

default `None`

size Controls the size of the pagination through CSS. Defaults to being normal sized.

One of the following:

- `'small'`
- `'large'`

default `None`

extra Any extra page parameters.

default `None`

parameter_name Name of the paging URL parameter.

default `'page'`

Usage:

```
{% bootstrap_pagination page %}
```

Example:

```
{% bootstrap_pagination lines url="/pagination?page=1" size="large" %}
```

4.12 bootstrap_jquery_url

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_jquery_url`**`()`

Tag name:

```
bootstrap_jquery_url
```

Return the full url to jQuery plugin to use

Default value: `https://code.jquery.com/jquery-3.2.1.min.js`

This value is configurable, see Settings section

Usage:

```
{% bootstrap_jquery_url %}
```

Example:

```
{% bootstrap_jquery_url %}
```

4.13 bootstrap_javascript_url

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_javascript_url`**`()`

Return the full url to the Bootstrap JavaScript library

Default value: `None`

This value is configurable, see Settings section

Tag name:

```
bootstrap_javascript_url
```

Usage:

```
{% bootstrap_javascript_url %}
```

Example:

```
{% bootstrap_javascript_url %}
```

4.14 bootstrap_css_url

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_css_url`**`()`

Return the full url to the Bootstrap CSS library

Default value: `None`

This value is configurable, see Settings section

Tag name:

```
bootstrap_css_url
```

Usage:

```
{% bootstrap_css_url %}
```

Example:

```
{% bootstrap_css_url %}
```

4.15 bootstrap_css

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_css`**`()`

Return HTML for Bootstrap CSS. Adjust url in settings. If no url is returned, we don't want this statement to return any HTML. This is intended behavior.

Default value: `None`

This value is configurable, see Settings section

Tag name:

```
bootstrap_css
```

Usage:

```
{% bootstrap_css %}
```

Example:

```
{% bootstrap_css %}
```

4.16 bootstrap_javascript

`bootstrap4.templatetags.bootstrap4.`**`bootstrap_javascript`**`(`*jquery=u'falsy'*`)`

Return HTML for Bootstrap JavaScript.

Adjust url in settings. If no url is returned, we don't want this statement to return any HTML. This is intended behavior.

Default value: `falsy`

This value is configurable, see Settings section

Tag name:

```
bootstrap_javascript
```

Parameters:

jquery falsy|slim|full (default=falsy)

Usage:

```
{% bootstrap_javascript %}
```

Example:

```
{% bootstrap_javascript jquery="slim" %}
```

CHAPTER 5

Settings

The django-bootstrap4 has some pre-configured settings.

They can be modified by adding a dict variable called `BOOTSTRAP4` in your `settings.py` and customizing the values you want;

The `BOOTSTRAP4` dict variable contains these settings and defaults:

```
# Default settings
BOOTSTRAP4 = {

    # The URL to the jQuery JavaScript file
    'jquery_url': '//code.jquery.com/jquery.min.js',

    # The Bootstrap base URL
    'base_url': '//maxcdn.bootstrapcdn.com/bootstrap/3.3.7/',

    # The complete URL to the Bootstrap CSS file (None means derive it from base_url)
    'css_url': None,

    # The complete URL to the Bootstrap CSS file (None means no theme)
    'theme_url': None,

    # The complete URL to the Bootstrap JavaScript file (None means derive it from␣
↪base_url)
    'javascript_url': None,

    # Put JavaScript in the HEAD section of the HTML document (only relevant if you␣
↪use bootstrap4.html)
    'javascript_in_head': False,

    # Include jQuery with Bootstrap JavaScript (affects django-bootstrap4 template␣
↪tags)
    'include_jquery': False,

    # Label class to use in horizontal forms
    'horizontal_label_class': 'col-md-3',
    # Field class to use in horizontal forms
    'horizontal_field_class': 'col-md-9',

    # Set placeholder attributes to label if no placeholder is provided
    'set_placeholder': True,

    # Class to indicate required (better to set this in your Django form)
    'required_css_class': '',

    # Class to indicate error (better to set this in your Django form)
    'error_css_class': 'has-error',

    # Class to indicate success, meaning the field has valid input (better to set␣
↪this in your Django form)
    'success_css_class': 'has-success',

    # Renderers (only set these if you have studied the source and understand the␣
↪inner workings)
    'formset_renderers':{
        'default': 'bootstrap4.renderers.FormsetRenderer',
    },
    'form_renderers': {
        'default': 'bootstrap4.renderers.FormRenderer',
    },
    'field_renderers': {
        'default': 'bootstrap4.renderers.FieldRenderer',
        'inline': 'bootstrap4.renderers.InlineFieldRenderer',
    },
}
```

CHAPTER 6

Templates

You can customize the output of `django-bootstrap4` by writing your own templates. These templates are available:

6.1 bootstrap4/field_help_text_and_errors.html

This renders the help text and error of each field.

Variable `help_text_and_errors` contains an array of strings.

6.2 bootstrap4/form_errors.html

This renders the non field errors of a form.

Variable `errors` contains an array of strings.

6.3 bootstrap4/messages.html

This renders the Django messages variable.

Variable `messages` contains the messages as described in https://docs.djangoproject.com/en/dev/ref/contrib/messages/#displaying-messages

`messages` is passed through three built-in filters

safe <https://docs.djangoproject.com/en/dev/ref/templates/builtins/#std:templatefilter-safe>

urlize <https://docs.djangoproject.com/en/dev/ref/templates/builtins/#std:templatefilter-urlize>

linebreaksbr <https://docs.djangoproject.com/en/dev/ref/templates/builtins/#std:templatefilter-linebreaksbr>

6.4 Other

There are two more templates, `bootstrap4/bootstrap4.html` and `bootstrap4/pagination.html`. You should consider these private for now, meaning you can use them but not modify them.

CHAPTER 7

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

7.1 Types of Contributions

7.1.1 Report Bugs

Report bugs at https://github.com/zostera/django-bootstrap4/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

7.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

7.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

7.1.4 Write Documentation

django-bootstrap4 could always use more documentation, whether as part of the official django-bootstrap4 docs, in docstrings, or even on the web in blog posts, articles, and such.

7.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/zostera/django-bootstrap4/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

7.2 Get Started!

Ready to contribute? Here's how to set up *django-bootstrap4* for local development.

1. Fork the *django-bootstrap4* repo on GitHub.
2. Clone your fork locally:

   ```
   $ git clone https://github.com/your_username_here/django-bootstrap4.git
   ```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

   ```
   $ mkvirtualenv django-bootstrap4
   $ cd django-bootstrap4/
   $ python setup.py develop
   ```

4. Create a branch for local development:

   ```
   $ git checkout -b name-of-your-bugfix-or-feature
   ```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 bootstrap4 tests
$ python setup.py test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

   ```
   $ git add .
   $ git commit -m "Your detailed description of your changes."
   $ git push origin name-of-your-bugfix-or-feature
   ```

7. Submit a pull request through the GitHub website.

7.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, and 3.3, and for PyPy. Check https://travis-ci.org/dyve/django-bootstrap4/pull_requests and make sure that the tests pass for all supported Python versions.

7.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_bootstrap4
```

CHAPTER 8

Authors

This application is developed and maintained by Zostera.

8.1 Development Lead

- Dylan Verheul <dylan@dyve.net>

8.2 Contributors

- Allard Stijnman <a.g.stijnman@gmail.com>
- Austin Whittier <austin.whitt@gmail.com>
- Caio Ariede <caio.ariede@gmail.com>
- Fabio C. Barrionuevo da Luz <bnafta@gmail.com>
- Fabio Perfetti <perfabio87@gmail.com>
- Jay Pipes <jaypipes@gmail.com>
- Jonas Hagstedt <hagstedt@gmail.com>
- Jordan Starcher <jstarcher@gmail.com>
- Juan Carlos <juancarlospaco@gmail.com>
- Markus Holtermann <info@markusholtermann.eu>
- Nick S <nsmith448@gmail.com>
- Owais Lone <loneowais@gmail.com>
- pmav99 <pmav99@users.noreply.github.com>
- Richard Hajdu <tuskone16@gmail.com>

- Timothy Allen <tim@pyphilly.org>

CHAPTER 9

History

9.1 0.0.5 (2018-01-03)

- Drop develop branch, work with master and feature branches
- Clean up history file
- Upgrade Bootstrap to 4.0.0-beta.3
- Use *col-4* rather than *col-xs-4* (#54)
- Added pagination alignment options (#56)
- Fixed form field errors and help texts (#39)
- Use django language if `USE_I18N=True` (#53)

9.2 0.0.4 (2017-10-25)

- Upgrade Bootstrap to 4.0.0-beta.2
- Fix settings, tags and tests for remote JS and CSS

9.3 0.0.3 (2017-09-22)

- Upgraded bootstrap4 to the beta version

9.4 0.0.2 (2017-09-06)

- Upgraded jQuery version from CDN to 3.2.1 (#17)

- Added proper pagination layout classes (#19)

9.5 0.0.1 (2017-06-04)

- First release

9.6 Unreleased (2017-04-25)

- Remove *bootstrap_icon*, BS4 no longer has default icons.
- Various changes to get from 3 to 4, started MIGRATE.rst.
- Started *django-bootstrap4* based on *django-bootstrap3*.
- Thanks everybody that contributed to *django-bootstrap3*!

Index