fluentcms-jumbotron Documentation

Release 1.0

Diederik van der Boor

September 11, 2016

Contents

1 Installation **3**

2 Frontend styling **5**
2.1 Customizing 5
2.2 Full page width 5
2.3 Contributing 6

Displaying a Bootstrap 3 Jumbotron in a page

CHAPTER 1

Installation

First install the module, preferably in a virtual environment. It can be installed from PyPI:

```
pip install fluentcms-jumbotron
```

First make sure the project is configured for django-fluent-contents.

Then add the following settings:

```
INSTALLED_APPS += (
    'fluentcms_jumbotron',
)

FLUENT_CONTENTS_PLACEHOLDER_CONFIG = {
    'slot name': {
        'plugins': ('JumbotronPlugin', ...),
    },
}
```

The database tables can be created afterwards:

```
./manage.py migrate
```

CHAPTER 2

Frontend styling

The jumbotron is rendered with the HTML that Bootstrap prescribes:

```
<div class="jumbotron">
  <h1>Hello, world!</h1>
  <p>...</p>
  <p><a class="btn btn-primary btn-lg" href="#" role="button">Learn more</a></p>
</div>
```

The standard Bootstrap 3 CSS will provide a reasonable styling for this, which can either be overwritten, or replaced in your own CSS files. The defaults provided by Bootstap 3 is: https://github.com/twbs/bootstrap-sass/blob/master/assets/stylesheets/bootstrap/_jumbotron.scss

2.1 Customizing

Centering, adding backgrounds, etc.. all happen by adding CSS lines. For example:

```
.jumbotron {
  background: url('/static/frontend/images/background.jpg') no-repeat fixed 0 0;
  background-size: cover;
  color: #fff;
  text-align: center;
}

.jumbotron .btn {
  margin-top: 12px;  /* For Sass: $padding-base-vertical * 2; */
}
```

When you use Sass, you can also override the Sass variables.

2.2 Full page width

To display the Bootstrap Jumbotron full page, you likely need to break out of the container the `JumbotronPlugin` is rendered in. For example, when your page looks like:

```
<div class="container">
    {% page_placeholder "homepage" title="Homepage" role="m" %}
</div>
```

You can change that into:

```
<div class="container">
    {% page_placeholder "homepage" title="Homepage" role="m" template="pages/placeholders/homepage.ht
</div>
```

The `pages/placeholders/homepage.html` template looks like:

```
{% for contentitem, html in contentitems %}
  {% if contentitem.plugin.name == 'JumbotronPlugin' %}
    </div>
    {{ html }}
    <div class="container">
  {% else %}
    {{ html }}
  {% endif %}
{% endfor %}
```

Note the exact HTML tags depend on your frontend HTML layout.

The `cachable=1` flag is a promise that the template always returns the same result for every request. Otherwise, remove it.

2.3 Contributing

If you like this module, forked it, or would like to improve it, please let us know! Pull requests are welcome too. :-)