
Django CKEditor Documentation

Release 6.7.0

The Django-CKEditor Team

Sep 21, 2023

Contents

1	Installation	5
1.1	Required	5
1.2	Required for using widget with file upload	6
2	Optional - customizing CKEditor editor	7
2.1	Optional for file upload	8
3	Usage	11
3.1	Field	11
3.2	Widget	11
3.3	Outside of django admin	12
3.4	Management Commands	13
3.5	Using S3	13
3.6	If you want to use allowedContent	13
4	Plugins:	15
5	Restricting file upload	17
6	Demo / Test application	19
7	Running selenium test	21
8	Troubleshooting	23
9	Example ckeditor configuration	25
10	AUTHORS	27
10.1	Created By	27
10.2	Contributors	27
11	Changelog	29
11.1	Unreleased	29
11.2	6.7.0	29
11.3	6.6.0	29
11.4	6.5.0	29
11.5	6.4.0	30
11.6	6.3.0	30

11.7	6.2.0	30
11.8	6.1.0	30
11.9	6.0.0	30
11.10	5.9.0	31
11.11	5.8.0	31
11.12	5.7.1	31
11.13	5.7.0	31
11.14	5.6.1	31
11.15	5.6.0	31
11.16	5.5.0	32
11.17	5.4.0	32
11.18	5.3.1	32
11.19	5.3.0	32
11.20	5.2.2	32
11.21	5.2.1	33
11.22	5.2.0	33
11.23	5.1.1	33
11.24	5.1.0	33
11.25	5.0.3	33
11.26	5.0.2	33
11.27	5.0.1	33
11.28	5.0.0 (4.5.3)	34
11.29	4.5.1	34
11.30	4.5.0	34
11.31	4.4.8	34
11.32	4.4.7	35
11.33	4.4.6	35
11.34	4.4.5	35
11.35	4.4.4	35
11.36	4.4.0	35
11.37	4.2.8	35
11.38	4.2.7	36
11.39	4.2.5 / 4.2.6	36
11.40	4.2.4	36
11.41	4.2.3	36
11.42	4.2.2	36
11.43	4.2.1	36
11.44	4.0.2	36
11.45	3.6.2.1	37
11.46	3.6.2	37
11.47	0.0.9	37
11.48	0.0.8	37
11.49	0.0.7	37
11.50	0.0.6	37
11.51	0.0.5	38
11.52	0.0.4	38
11.53	0.0.3	38
11.54	0.0.2	38
11.55	0.0.1	38

Django admin CKEditor integration.

Provides a `RichTextField`, `RichTextUploadingField`, `CKEditorWidget` and `CKEditorUploadingWidget` utilizing CKEditor with image uploading and browsing support included.

This version also includes:

1. support to django-storages (works with S3)
2. updated ckeditor to version 4.18.0
3. included all ckeditor language and plugin files to make everyone happy! ([only the plugins maintained by the ckeditor develops team](#))

Contents

- *Django CKEditor*
 - *Installation*
 - * *Required*
 - * *Required for using widget with file upload*
 - *Optional - customizing CKEditor editor*
 - * *Optional for file upload*
 - *Usage*
 - * *Field*
 - * *Widget*
 - * *Outside of django admin*
 - * *Management Commands*
 - * *Using S3*
 - * *If you want to use allowedContent*
 - *Plugins:*
 - *Restricting file upload*
 - *Demo / Test application*
 - *Running selenium test*
 - *Troubleshooting*
 - *Example ckeditor configuration*
- *AUTHORS*
 - *Created By*

- *Contributors*
- *Changelog*
 - *Unreleased*
 - *6.7.0*
 - *6.6.0*
 - *6.5.0*
 - *6.4.0*
 - *6.3.0*
 - *6.2.0*
 - *6.1.0*
 - *6.0.0*
 - *5.9.0*
 - *5.8.0*
 - *5.7.1*
 - *5.7.0*
 - *5.6.1*
 - *5.6.0*
 - *5.5.0*
 - *5.4.0*
 - *5.3.1*
 - *5.3.0*
 - *5.2.2*
 - *5.2.1*
 - *5.2.0*
 - *5.1.1*
 - *5.1.0*
 - *5.0.3*
 - *5.0.2*
 - *5.0.1*
 - *5.0.0 (4.5.3)*
 - *4.5.1*
 - *4.5.0*
 - *4.4.8*
 - *4.4.7*
 - *4.4.6*

- 4.4.5
- 4.4.4
- 4.4.0
- 4.2.8
- 4.2.7
- 4.2.5 / 4.2.6
- 4.2.4
- 4.2.3
- 4.2.2
- 4.2.1
- 4.0.2
- 3.6.2.1
- 3.6.2
- 0.0.9
- 0.0.8
- 0.0.7
- 0.0.6
- 0.0.5
- 0.0.4
- 0.0.3
- 0.0.2
- 0.0.1

1.1 Required

1. Install or add django-ckeditor to your python path.

```
pip install django-ckeditor
```

2. Add ckeditor to your INSTALLED_APPS setting.
3. Run the collectstatic management command: `$./manage.py collectstatic`. This will copy static CKEditor required media resources into the directory given by the `STATIC_ROOT` setting. See [Django's documentation on managing static files](#) for more info.
4. CKEditor needs to know where its assets are located because it loads them lazily only when needed. The location is determined in the `ckeditor-init.js` script. and defaults to `static/ckeditor/ckeditor/`. This does not work all the time, for example when using `ManifestStaticFilesStorage`, any asset packaging pipeline or whatnot. `django-ckeditor` is quite good at automatically detecting the correct place even then, but sometimes you have to hardcode `CKEDITOR_BASEPATH` somewhere. This can be hardcoded in settings, i.e.:

```
CKEDITOR_BASEPATH = "/my_static/ckeditor/ckeditor/"
```

It is possible to override the `admin/change_form.html` template with your own if you really need to do this, i.e.:

```
{% extends "admin/change_form.html" %}

{% block extrahead %}
<script>window.CKEDITOR_BASEPATH = '/my_static/ckeditor/ckeditor/';</script>
{% block.super %}
{% endblock %}
```

Of course, you should adapt this snippet to your needs when using CKEditor outside the admin app.

1.2 Required for using widget with file upload

1. Add `ckeditor_uploader` to your `INSTALLED_APPS` setting.
2. Add a `CKEDITOR_UPLOAD_PATH` setting to the project's `settings.py` file. This setting specifies a relative path to your CKEditor media upload directory. CKEditor uses Django's storage API. By default, Django uses the file system storage backend (it will use your `MEDIA_ROOT` and `MEDIA_URL`) and if you don't use a different backend you have to have write permissions for the `CKEDITOR_UPLOAD_PATH` path within `MEDIA_ROOT`, i.e.:

```
CKEDITOR_UPLOAD_PATH = "uploads/"
```

When using default file system storage, images will be uploaded to “uploads” folder in your `MEDIA_ROOT` and urls will be created against `MEDIA_URL` (`/media/uploads/image.jpg`).

If you want to be able to have control over filename generation, you have to add a custom filename generator to your settings:

```
# utils.py

def get_filename(filename, request):
    return filename.upper()
```

```
# settings.py

CKEDITOR_FILENAME_GENERATOR = 'utils.get_filename'
```

CKEditor has been tested with django `FileSystemStorage` and `S3BotoStorage`. There are issues using `S3Storage` from `django-storages`.

3. For the default filesystem storage configuration, `MEDIA_ROOT` and `MEDIA_URL` must be set correctly for the media files to work (like those uploaded by the ckeditor widget).
4. Add CKEditor URL include to your project's `urls.py` file:

```
path('ckeditor/', include('ckeditor_uploader.urls')),
```

5. Note that by adding those URLs you add views that can upload and browse through uploaded images. Since django-ckeditor 4.4.6, those views are decorated using `@staff_member_required`. If you want a different permission decorator (`login_required`, `user_passes_test` etc.) then add views defined in `ckeditor.urls` manually to your `urls.py`.

Optional - customizing CKEditor editor

1. Add a CKEDITOR_CONFIGS setting to the project's `settings.py` file. This specifies sets of CKEditor settings that are passed to CKEditor (see CKEditor's [Setting Configurations](#)), i.e.:

```
CKEDITOR_CONFIGS = {
    'awesome_ckeditor': {
        'toolbar': 'Basic',
    },
}
```

The name of the settings can be referenced when instantiating a `RichTextField`:

```
content = RichTextField(config_name='awesome_ckeditor')
```

The name of the settings can be referenced when instantiating a `CKEditorWidget`:

```
widget = CKEditorWidget(config_name='awesome_ckeditor')
```

By specifying a set named `default` you'll be applying its settings to all `RichTextField` and `CKEditorWidget` objects for which `config_name` has not been explicitly defined

```
CKEDITOR_CONFIGS = {
    'default': {
        'toolbar': 'full',
        'height': 300,
        'width': 300,
    },
}
```

It is possible to create a custom toolbar

```
CKEDITOR_CONFIGS = {
    'default': {
        'toolbar': 'Custom',
        'toolbar_Custom': [
```

(continues on next page)

(continued from previous page)

```
        ['Bold', 'Italic', 'Underline'],
        ['NumberedList', 'BulletedList', '-', 'Outdent', 'Indent', '-'],
        → 'JustifyLeft', 'JustifyCenter', 'JustifyRight', 'JustifyBlock'],
        ['Link', 'Unlink'],
        ['RemoveFormat', 'Source']
    ]
}
}
```

If you want or need plugins which are not part of django-ckeditor’s plugin set you may specify assets and plugins as follows:

```
text = RichTextField(
    config_name='forum-post',

    # CKEDITOR.config.extraPlugins:
    extra_plugins=['someplugin'],

    # CKEDITOR.plugins.addExternal(...)
    external_plugin_resources=[(
        'someplugin',
        '/static/.../path-to-someplugin/',
        'plugin.js',
    )],
)

Alternatively, those settings can also be provided through
`CKEDITOR_CONFIGS`.
```

2.1 Optional for file upload

1. All uploaded files are slugified by default. To disable this feature, set `CKEDITOR_UPLOAD_SLUGIFY_FILENAME` to `False`.
2. Set the `CKEDITOR_RESTRICT_BY_USER` setting to `True` in the project’s `settings.py` file (default `False`). This restricts access to uploaded images to the uploading user (e.g. each user only sees and uploads their own images). Upload paths are prefixed by the string returned by `get_username`. If `CKEDITOR_RESTRICT_BY_USER` is set to a string, the named property is used instead. Superusers can still see all images. **NOTE:** This restriction is only enforced within the CKEditor media browser.
3. Set the `CKEDITOR_BROWSE_SHOW_DIRS` setting to `True` to show directories on the “Browse Server” page. This enables image grouping by directory they are stored in, sorted by date.
4. Set the `CKEDITOR_RESTRICT_BY_DATE` setting to `True` to bucket uploaded files by year/month/day.
5. You can set a custom file storage for CKEditor uploader by defining it under `CKEDITOR_STORAGE_BACKEND` variable in settings.
6. You can set `CKEDITOR_IMAGE_BACKEND` to one of the supported backends to enable thumbnails in ckeditor gallery. By default, no thumbnails are created and full-size images are used as preview. Supported backends:
 - `ckeditor_uploader.backends.PillowBackend`: Uses Pillow
7. With the `PillowBackend` backend, you can change the thumbnail size with the `CKEDITOR_THUMBNAIL_SIZE` setting (formerly `THUMBNAIL_SIZE`). Default value: (75, 75)

8. With the `PillowBackend` backend, you can convert and compress the uploaded images to jpeg, to save disk space. Set the `CKEDITOR_FORCE_JPEG_COMPRESSION` setting to `True` (default `False`) You can change the `CKEDITOR_IMAGE_QUALITY` setting (formerly `IMAGE_QUALITY`), which is passed to Pillow:

The image quality, on a scale from 1 (worst) to 95 (best). The default is 75. Values above 95 should be avoided; 100 disables portions of the JPEG compression algorithm and results in large files with hardly any gain in image quality.

This feature is disabled for animated images.

3.1 Field

The quickest way to add rich text editing capabilities to your models is to use the included `RichTextField` model field type. A CKEditor widget is rendered as the form field but in all other regards the field behaves like the standard Django `TextField`. For example:

```
from django.db import models
from ckeditor.fields import RichTextField

class Post(models.Model):
    content = RichTextField()
```

For file upload support use `RichTextUploadingField` from `ckeditor_uploader.fields`.

3.2 Widget

Alternatively, you can use the included `CKEditorWidget` as the widget for a formfield. For example:

```
from django import forms
from django.contrib import admin
from ckeditor.widgets import CKEditorWidget

from post.models import Post

class PostAdminForm(forms.ModelForm):
    content = forms.CharField(widget=CKEditorWidget())
    class Meta:
        model = Post
        fields = '__all__'

class PostAdmin(admin.ModelAdmin):
```

(continues on next page)

(continued from previous page)

```
form = PostAdminForm

admin.site.register(Post, PostAdmin)
```

For file upload support use `CKEditorUploadingWidget` from `ckeditor_uploader.widgets`.

Overriding widget template

In Django `>=1.11` for overriding `ckeditor/widget.html` you have three ways:

1. Place `ckeditor/widget.html` in `BASE_DIR/templates`

- Change `FORM_RENDERER` to `TemplateSettings`.

```
FORM_RENDERER = 'django.forms.renderers.TemplateSetting'
```

- Include `templates` folder in `DIRS`

```
TEMPLATES = [{
    ...
    'DIRS': [os.path.join(BASE_DIR, 'templates'), ],
    ...
}]
```

- Add `'django.forms'` to `INSTALLED_APPS`.

2. Place `ckeditor/widget.html` in `your_app/templates` and place `'your_app'` **before** `'ckeditor'` and `'ckeditor_uploader'` in `INSTALLED_APPS`.

3. Inherit from `CKEditorWidget` and override `template_name` with a custom template available in `TEMPLATES DIRS` as defined `settings.py`.

```
class MyCustomCKEditorWidget(CKEditorWidget):
    template_name = "templates/custom_ckeditor/widget.html"
```

3.3 Outside of django admin

When you are rendering a form outside the admin panel, you'll have to make sure all form media is present for the editor to work. One way to achieve this is like this:

```
<form>
    {{ myform.media }}
    {{ myform.as_p }}
    <input type="submit"/>
</form>
```

or you can load the media manually as it is done in the demo app:

```
{% load static %}
<script type="text/javascript" src="{% static "ckeditor/ckeditor-init.js" %}"></script>
<script type="text/javascript" src="{% static "ckeditor/ckeditor/ckeditor.js" %}"></script>
```

When you need to render `RichTextField`'s HTML output in your templates safely, just use `{{ content|safe }}`, Django's safe filter

3.4 Management Commands

Included is a management command to create thumbnails for images already contained in CKEDITOR_UPLOAD_PATH. This is useful to create thumbnails when using django-ckeditor with existing images. Issue the command as follows:

```
$ ./manage.py generateckeditorthumbnails
```

NOTE: If you're using custom views remember to include ckeditor.js in your form's media either through `{{ form.media }}` or through a `<script>` tag. Admin will do this for you automatically. See [Django's Form Media docs](#) for more info.

3.5 Using S3

See <https://django-storages.readthedocs.org/en/latest/>

NOTE: django-ckeditor will not work with S3 through django-storages without this line in settings.py:

```
AWS_QUERYSTRING_AUTH = False
```

3.6 If you want to use allowedContent

To get allowedContent to work, disable **stylesheetparser** plugin. So include this in your settings.py.:

```
CKEDITOR_CONFIGS = {  
    "default": {  
        "removePlugins": "stylesheetparser",  
    }  
}
```


CHAPTER 4

Plugins:

django-ckeditor includes the following ckeditor plugins, but not all are enabled by default:

```
allyhelp, about, adobeair, ajax, autoembed, autogrow, autolink, bbcode, clipboard, ↵  
↵codesnippet,  
codesnippetgeshi, colordialog, devtools, dialog, div, divarea, docprops, embed, ↵  
↵embedbase,  
embedsemantic, filetools, find, flash, forms, iframe, iframedialog, image, image2, ↵  
↵language,  
lineutils, link, liststyle, magicline, mathjax, menubutton, notification, ↵  
↵notificationaggregator,  
pagebreak, pastefromword, placeholder, preview, scayt, sharedspace, showblocks, ↵  
↵smiley,  
sourcedialog, specialchar, stylesheetparser, table, tableresize, tabletools, ↵  
↵templates, uicolor,  
uploadimage, uploadwidget, widget, wsc, xml
```

The image/file upload feature is done by the *uploadimage* plugin.

Restricting file upload

1. To restrict upload functionality to image files only, add `CKEDITOR_ALLOW_NONIMAGE_FILES = False` in your `settings.py` file. Currently non-image files are allowed by default.
2. By default the upload and browse URLs use `staff_member_required` decorator - `ckeditor_uploader/urls.py` - if you want other decorators just insert two urls found in that `urls.py` and don't include it.

CHAPTER 6

Demo / Test application

If you clone the repository you will be able to run the `ckeditor_demo` application.

1. `pip install -r ckeditor_demo_requirements.txt`
2. Run `python manage.py migrate`
3. Create a superuser if you want to test the widget in the admin panel
4. Start the development server.

There is a `forms.Form` on the main page (/) and a model in admin that uses the widget for a model field. Database is set to `sqlite3` and `STATIC/MEDIA_ROOT` to folders in temporary directory.

CHAPTER 7

Running selenium test

The recommended way to run selenium tests is using tox. Select the appropriate selenium driver using the `SELENIUM` environment variable and optionally specify that you want to run only one environment since selenium takes some time and/or since you do not have all supported versions of Python installed locally. The example uses the combination of Python 3.9 and Django 4.0 which is a supported combination at the time of writing:

```
# Either
SELENIUM=firefox tox -e py39-dj40

# Or
SELENIUM=chromium tox -e py39-dj40

# Or even
SELENIUM=firefox tox
```


CHAPTER 8

Troubleshooting

If your browser has problems displaying uploaded images in the image upload window you may need to change Django settings:

```
X_FRAME_OPTIONS = 'SAMEORIGIN'
```

More on <https://docs.djangoproject.com/en/1.11/ref/clickjacking/#setting-x-frame-options-for-all-responses>

Example ckeditor configuration

```
CKEDITOR_CONFIGS = {
  'default': {
    'skin': 'moono',
    # 'skin': 'office2013',
    'toolbar_Basic': [
      ['Source', '-', 'Bold', 'Italic']
    ],
    'toolbar_YourCustomToolbarConfig': [
      {'name': 'document', 'items': ['Source', '-', 'Save', 'NewPage', 'Preview',
↪ '-', 'Print', '-', 'Templates']},
      {'name': 'clipboard', 'items': ['Cut', 'Copy', 'Paste', 'PasteText',
↪ 'PasteFromWord', '-', 'Undo', 'Redo']},
      {'name': 'editing', 'items': ['Find', 'Replace', '-', 'SelectAll']},
      {'name': 'forms',
        'items': ['Form', 'Checkbox', 'Radio', 'TextField', 'Textarea', 'Select',
↪ 'Button', 'ImageButton',
        'HiddenField']},
      '/',
      {'name': 'basicstyles',
        'items': ['Bold', 'Italic', 'Underline', 'Strike', 'Subscript',
↪ 'Superscript', '-', 'RemoveFormat']},
      {'name': 'paragraph',
        'items': ['NumberedList', 'BulletedList', '-', 'Outdent', 'Indent', '-',
↪ 'Blockquote', 'CreateDiv', '-',
        'JustifyLeft', 'JustifyCenter', 'JustifyRight', 'JustifyBlock',
↪ '-', 'BidiLtr', 'BidiRtl',
        'Language']},
      {'name': 'links', 'items': ['Link', 'Unlink', 'Anchor']},
      {'name': 'insert',
        'items': ['Image', 'Flash', 'Table', 'HorizontalRule', 'Smiley',
↪ 'SpecialChar', 'PageBreak', 'Iframe']},
      '/',
      {'name': 'styles', 'items': ['Styles', 'Format', 'Font', 'FontSize']},
      {'name': 'colors', 'items': ['TextColor', 'BGColor']},
    ]
  }
}
```

(continues on next page)

(continued from previous page)

```

        {'name': 'tools', 'items': ['Maximize', 'ShowBlocks']},
        {'name': 'about', 'items': ['About']},
        '/', # put this to force next toolbar on new line
        {'name': 'yourcustomtools', 'items': [
            # put the name of your editor.ui.addButton here
            'Preview',
            'Maximize',

            ]},
    ],
    'toolbar': 'YourCustomToolbarConfig', # put selected toolbar config here
    # 'toolbarGroups': [{ 'name': 'document', 'groups': [ 'mode', 'document',
↪ 'doctools' ] }],
    # 'height': 291,
    # 'width': '100%',
    # 'filebrowserWindowHeight': 725,
    # 'filebrowserWindowWidth': 940,
    # 'toolbarCanCollapse': True,
    # 'mathJaxLib': '//'cdn.mathjax.org/mathjax/2.2-latest/MathJax.js?config=TeX-
↪ AMS_HTML',
    'tabSpaces': 4,
    'extraPlugins': ','.join([
        'uploadimage', # the upload image feature
        # your extra plugins here
        'div',
        'autolink',
        'autoembed',
        'embedsemantic',
        'autogrow',
        # 'devtools',
        'widget',
        'lineutils',
        'clipboard',
        'dialog',
        'dialogui',
        'elementspath'
    ]),
    }
}

```

10.1 Created By

1. shaunsephton

10.2 Contributors

1. riklaunim
2. 3point2
3. buchuki
4. chr15m
5. hedleyroos
6. jeffh
7. lihan
8. loop0
9. mwcx
10. tomwys
11. snbuback
12. And others <https://github.com/django-ckeditor/django-ckeditor/graphs/contributors>

11.1 Unreleased

11.2 6.7.0

1. CKEditor 4.22.1
2. Dark mode fixes.
3. Added support for Pillow 10.

11.3 6.6.0

1. Required a newer version of `django-js-asset` which actually works with Django 4.1.
2. CKEditor 4.21.0
3. Fixed the CKEditor styles when used with the dark Django admin theme.

11.4 6.5.0

1. Avoided calling `static()` if `CKEDITOR_BASEPATH` is defined.
2. Fixed `./manage.py generateckeditorthumbnails` to work again after the image uploader backend rework.
3. CKEditor 4.19.1
4. Stopped calling `static()` during application startup.
5. Added Django 4.1

11.5 6.4.0

1. Changed the context for the widget to deviate less from Django. Removed a few template variables which are not used in the bundled `ckeditor/widget.html` template. This only affects you if you are using a customized widget or widget template.
2. Dropped support for Python < 3.8, Django < 3.2.
3. Added a pre-commit configuration.
4. Removed the Travis CI configuration; Travis CI hasn't run our unit tests for months now.
5. Added a GitHub action for running tests.
6. Made selenium tests require opt in using a `SELENIUM=firefox` or `SELENIUM=chromium` environment variable.

11.6 6.3.0

1. CKEditor 4.18.0
2. Made it possible to override the CKEditor template in the widget class.
3. Changed `CKEDITOR_IMAGE_BACKEND` to require dotted module paths (the old identifiers are still supported for now).

11.7 6.2.0

1. CKEditor 4.17.1

11.8 6.1.0

1. CKEditor 4.16.1

11.9 6.0.0

1. Replace `ugettext_lazy()` with `gettext_lazy()`
2. CKEditor 4.14.1
3. Changed our JS script to listen for Django's `formset:added` signals instead of detecting clicks on inline buttons. This should fix compatibility with various Django admin skins.
4. Dropped compatibility guarantees for Django<2.2 and Python<3.6.
5. Reformatted the code using black, isort.
6. Added testing using Django 3.1.

11.10 5.9.0

1. Django 3.0 support
2. Python 3.8 support
3. Replace *staticfiles* templatetags library usage with *static*
4. Add a templates validation step to the tests
5. Internationalize ckeditor_upload *browse.html* template.
6. Add ckeditor_upload features and custom configuration example to *ckeditor_demo*
7. CKEditor 4.13.1

11.11 5.8.0

1. CKEditor 4.13

11.12 5.7.1

1. CKEditor 4.11.4
2. Fix JS handling again
3. Allow using settings to configure `extra_plugins` and `external_plugin_resources`

11.13 5.7.0

1. Fix Django 1.8 - 1.10 regression
2. Drop leftover support for Django older than 1.8
3. Django 2.2 support
4. Documentation updates
5. Minor fixes to JS handling

11.14 5.6.1

1. Fix bad pypi package

11.15 5.6.0

1. Django 2.1 compatibility, minimal supported Django version is 1.11 LTS
2. Option to set custom django file backend for CKEditor uploader app.

11.16 5.5.0

1. CKEditor 4.9.2
2. Documentation improvements
3. Allow non-string properties of user for CKEDITOR_RESTRICT_BY_USER

11.17 5.4.0

1. Django 2.0 compatibility

11.18 5.3.1

1. Actually include the code which sets CKEDITOR_BASEPATH.
2. CKEditor 4.7.3

11.19 5.3.0

1. CKEditor 4.7
2. Fix storage problems by setting CKEDITOR_BASEPATH (hopefully for real this time)
3. Documentation updates
4. Added a CKEDITOR_RESTRICT_BY_DATE setting to add uploaded files into folders containing the current date.
5. Added a CKEDITOR_FILEICONS setting that allows overriding the icons used by Gallerific.
6. Added a CKEDITOR_FILENAME_GENERATOR setting which allows specifying a callable which mangles the filename of uploaded files.
7. Added THUMBNAIL_SIZE and IMAGE_QUALITY settings for the Pillow image backend.
8. Actually include static assets for ckeditor_uploader in the pip-installable package.
9. Removed CKEDITOR_JQUERY_URL and the jQuery dependency. The CKEditor activation now uses plain JavaScript. Dependencies are `JSON.parse` and `document.querySelectorAll` which are supported in practically all used browsers these days.
10. Fixed a bug where the CKEditor language was not set individually for each request.

11.20 5.2.2

1. Django 1.11 support
2. Drop South migrations
3. Fix storage problems by setting CKEDITOR_BASEPATH

11.21 5.2.1

1. Fix CKEditor package static path

11.22 5.2.0

1. Django 1.10 updates
2. Development dependencies bump
3. CKEditor 4.6.1
4. Paste image support
5. Fix for ManifestStaticFilesStorage

11.23 5.1.1

1. Re-add missing additional CkEditor plugins

11.24 5.1.0

1. Updated CkEditor to 4.5.10
2. Django 1.10 compatibility changes
3. Documentation updates

11.25 5.0.3

1. Fix file/directory browsing and searching
2. Editor width style fixes
3. Added CKEDITOR_BROWSE_SHOW_DIRS
4. Added CKEDITOR_ALLOW_NONIMAGE_FILES
5. Python 2.6 compatibility fix

11.26 5.0.2

1. Added template missing in the package

11.27 5.0.1

1. Update Readme with backward-incompatible changes

11.28 5.0.0 (4.5.3)

1. Moved file upload code to new Django application - ckeditor_uploader. *RichTextField* doesn't use file upload which have been moved to *RichTextUploadingField*.

File upload support have been moved to ckeditor_uploader. The urls are in ckeditor_uploader.urls while for file uploading widget you have to use RichTextUploadingField from ckeditor_uploader.fields instead of RichTextField from from ckeditor.fields.

1. Updated ckeditor to 4.5.3 (from <https://github.com/ckeditor/ckeditor-dev/tree/4.5.3>)
2. Added new plugins from ckeditor maintainers: adobeair, ajax, autoembed, autogrow, autolink, bbcode, codesnippet, codesnippetgeshi, devtools, divarea, docprops, embed, embedbase, embedsemantic, filetools, iframedialog, image2, language, lineutils, mathjax, menubutton, notification, notificationaggregator, placeholder, sharedspace, sourcedialog, stylesheetparser, tablesize, uicolor, uploadimage, uploadwidget, widget, xml
3. Add `zip_safe=False` on setup config, to force does not create ".egg" file
4. Add python Wheel package configuration
5. Add setup.py functions to easy release ".egg" package and Wheel package, and tag version on git (`python setup.py publish` and `python setup.py tag`)
6. Improved Tox configuration to code coverage check, code quality check (flake8), imports order check (isort) and test with django master branch
7. Add code quality configurations
8. Add EditorConfig configuration file
9. Refactored code to be in compliance with PEP8

11.29 4.5.1

1. Fixed unbound variable in non-image file upload

11.30 4.5.0

1. Updated ckeditor to 4.5.1
2. Reverted django.contrib.staticfiles.templatetags.staticfiles.static usage causing problems with some storages
3. Allow non-image files to be upload (the upload widget expects images so the user experience isn't best at the moment)
4. Few refactors and fixes to selenium tests

11.31 4.4.8

1. Python 3 compatibility fixes
2. Get static files paths in a proper way
3. Fix Django 1.7 deprecation warning
4. More examples in readme

11.32 4.4.7

1. Allow only POST requests on upload view.
2. Exclude hidden files from image browser
3. Prevent caching of image browser view
4. Use lazy JSON encoder to support i18n in CKEditor settings.
5. Misc documentation updates
6. Check for jQuery presence correctly
7. Update to CKEditor 4.4.6

11.33 4.4.6

1. Make upload/browse views be `staff_member_required` by default (can be overridden)
2. Fix ckeditor initialisation code breaking with other jQuery versions.
3. Support grappelli inline form widgets.
4. Remove odd left margin from widget template.
5. Allow running selenium tests with chromium.

11.34 4.4.5

1. Post merge package name fix in Readme

11.35 4.4.4

1. Update CKEditor to 4.4.4 full package - for all plugins and static files you may need
2. Fixes for inline editor
3. Editor initialisation uses jQuery. You need to specify `CKEDITOR_JQUERY_URL` for it to work. You can use:

```
CKEDITOR_JQUERY_URL = '//ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js'
```

11.36 4.4.0

1. Update CKEditor to 4.4.1
2. Django 1.7 compatibility fix

11.37 4.2.8

1. Update CKEditor to 4.3.3

11.38 4.2.7

1. Fix slugifying to empty filename if only bad characters given in filename. Use random string as fallback.
2. Don't use IMG tags for non image files in ckeditor file browser.
3. Remove non-existing image reference from CSS files that broke collectstatic.
4. Misc fixes

11.39 4.2.5 / 4.2.6

1. Fix static files installation - switch from distutils to setuptools

11.40 4.2.4

1. Added new demo application with selenium integration test
2. tox setup for Python 3.3 and 2.7 testing
3. Extracted image processing to backends. PIL/Pillow is optional now. Other backends can be added.
4. Fixed a bug with thumbnail generation

11.41 4.2.3

1. Python 3.3 compatibility
2. All uploaded files are slugified by default (New settings CKEDITOR_SLUGIFY_FILENAME)
3. Upload file when editing a link (<a href>) now works properly

11.42 4.2.2

1. Python 3.3 compatibility in widgets.py

11.43 4.2.1

1. Include CKEditor version 4.2.1.
2. Support Django 1.6

11.44 4.0.2

1. Include CKEditor version 4.0.2.

11.45 3.6.2.1

1. Remove unwanted static files from distribution.
2. Use Pillow instead of PIL since it builds on all systems.

11.46 3.6.2

1. Include CKEditor version 3.6.2.
2. Initial work on Django aligned theme.
3. Fix schema slash removal issue on media url generation. Thanks [mwcz](#)
4. Added compatibility for South. Thanks [3point2](#)
5. Prevented settings from leaking between widget instances. Thanks [3point2](#)
6. Fixed config_name conflict when verbose_name is used as first positional argument for a field. Thanks [3point2](#)
7. Refactored views to allow use of file walking with local paths. Thanks [3point2](#)
8. Added command to generate thumbnails. Thanks [3point2](#)
9. Migrated from using media to static file management.

11.47 0.0.9

1. Added ability to configure CKeditor through a CKEDITOR_CONFIGS settings. Thanks [jeffh](#) for the input.

11.48 0.0.8

1. Removed buggy url include check.

11.49 0.0.7

1. Egg package corrected to exclude testing admin.py and models.py.

11.50 0.0.6

1. Enforce correct configuration.
2. Changed upload behavior to separate files into directories by upload date. Thanks [loop0](#) .
3. Added ability to limit user access to uploaded content (see the CKEDITOR_RESTRICT_BY_USER setting). Thanks [chr15m](#) for the input.
4. Added initial set of much needed tests.
5. General cleanup, light refactor.

11.51 0.0.5

1. csrf_exempt backwards compatability. Thanks [chr15m](#) .

11.52 0.0.4

1. Include resources, sorry about that.

11.53 0.0.3

1. More robust PIL import. Thanks [buchuki](#) .
2. Better CKEDITOR_MEDIA_PREFIX setting error.

11.54 0.0.2

1. Included README.rst in manifest.

11.55 0.0.1

1. Added CKEDITOR_UPLOAD_PREFIX setting. Thanks [chr15m](#) for the input.