twentytab-customadmin Documentation

Release 0.7

20tab srl

April 28, 2014

Contents

Contents:

If you’re new to twentytab-customadmin you may want to start with these documets to get you up and running:

CHAPTER 1

Overview

1.1 What is `twentytab-customadmin`?

`twentytab-customadmin` A django app to customize your admin interface with icons for applications and models and other (useful and beautiful) stuff.

CHAPTER 2

Installation

Contents

2.1 Official releases

Official releases are available from PyPI.

2.1.1 Install via pip

The easier way to install `customadmin` is via pip. Enter this command:

```
pip install twentytab-customadmin
```

...and the package will install automatically.

2.1.2 Install via source code

Alternatively, you can download the .gz distribution file available from PyPI and unpack it. Inside there is a script named `setup.py`. Enter this command:

```
python setup.py install
```

...and, again, the package will install automatically.

2.2 Development versions

If you prefer, you can get the latest source from our github repository:

```
git clone https://github.com/20tab/twentytab-customadmin.git
```

Add the resulting folder to your PYTHONPATH or symlink the `twentytab-customadmin` directory inside it into a directory which is on your PYTHONPATH, such as your Python installation's `site-packages` directory.

You can verify that the application is available on your PYTHONPATH by opening a Python interpreter and entering the following commands:

```
>>> import customadmin
>>> customadmin.VERSION
(0, 7)
```

When you want to update your copy of the source code, run `git pull` from within the `twentytab-customadmin` directory.

Caution: The development version may contain bugs which are not present in the release version and introduce backwards-incompatible changes.
If you're tracking master, keep an eye on the recent Commit History before you update your copy of the source code.

CHAPTER 3

Configuration

Contents

3.1 settings.py

Open `settings.py` and add `twentytab-customadmin` to your `INSTALLED_APPS` (before django.contrib.admin):

```
INSTALLED_APPS = {
    ...,
    'customadmin',
    'django.contrib.admin',
    'inspectmodel',
    'imagekit',
    'image_ui',
    'rosetta',
    ...
}
```

3.2 urls.py

Open `urls.py` and add those views to your `urlpatters`:

```
urlpatterns = patterns('',
    ... ,
    (r'^admin/', include(contrib.admin.site.urls)),
    (r'', include('customadmin.urls')),
    (r'^rosetta/', include('rosetta.urls')),
    (r'', include('inspectmodel.urls')),
    ...
)
```

3.3 Static files

Run collectstatic command or map static directory. If you use uWSGI you can map static files:

```
static-map = /static/customadmin/=%(path-to-site-packages)/customadmin/static/customadmin
static-map = /static/colorful/=%(path-to-site-packages)/colorful/static/colorful
static-map = /static/sortable/=%(path-to-site-packages)/sortable/static/sortable
static-map = /static/image_ui/=%(path-to-site-packages)/image_ui/static/image_ui
```

3.4 Run syncdb

Last (but not least) run:

```
python manage.py syncdb
```

To create `twentytab-customadmin` tables, now your are ready for the funny part.

CHAPTER 4

Usage

In your admin interface you now have `Customadmin` app with 4 models:

Contents

4.1 Custom Admin

As the name suggests this model is the core of this app, here you can customize the general aspect of the admin interface.

4.1.1 Branding

Branding: Set Branding. (default: Django Administration)

Branding link: Set Branding's link. (ex. '/' to redirect to the homepage) (default: No redirect)

Branding image: Set a logo to be displayed with or instead of the Branding. (default: No image)

Default: Set it if you want to enable this Custom Admin configuration.

4.1.2 View Option

View mode: This is the most important option. Set the interface that you prefer between 5 options:

> ————: Keep the default django admin interface with the options that you set in this configuration but you cannot use *Custom App*, *Custom Model* and *Custom Link*.
>
> *Use custom app system*: Like ———— but you can use *Custom App*, *Custom Model* and *Custom Link*.

> *Use apps' icon system*: Apps in the admin homepage are shown as icons intead of a list of links to models.
>
> *Use apps and models icon system*: Like *Use apps' icon system* but models too are shown as icons instead of a list of links.
>
> *Use models' icons system in index group models by app*: Models are shown as icons grouped by app in the admin homepage.
>
> *Use models' icons system in index ungroup models by app*: Like *Use models' icons system in index group models by app* but models are not grouped by app.

Use log sidebar: Check if you want the log sidebar in the admin homepage to be shown.

Autocomplete app: Uncheck if you want `customadmin` to hide the apps not defined in *Custom App*.

Autocomplete model: Uncheck if you want `customadmin` to hide the apps not defined in *Custom Model*.

4.1.3 Images

Default app image: Set an image that will replace the default app icon shown in the admin homepage. (this image will be applied to all apps, you can set different images per app in *Custom App*)

Default model image: Set an image that will replace the default model icon shown in the admin homepage. (this image will be applied to all models, you can set different images per model in *Custom Model*)

4.1.4 Style

Here you can change colors, font, font-size and so on to the admin interface. (default: if not changed it keeps django administration default style)

4.1.5 Code

Html head: Add extra code to the <head> tag. (ex. you can add some javascript)

Css code: Add extra css.

Use css code: if not checked the css code defined in **Css code** is disabled.

4.2 Custom App

`twentytab-customadmin` automatically creates one entry for each synchronized app.

You can modify the order in which apps are displayed in the admin homepage simply dragging and dropping them (thanks to twentytab-sortable)

change the **Verbose app name**

change the app's icon image (only if you selected *Use apps' icon system* or *Use apps and models icon system* in **View Mode** in your *Custom Admin* configuration)

or decide to hide this application from the admin homepage (only if **Autocompelte App** field in your *Custom Admin* configuration is unchecked).

4.3 Custom Model

Like *Custom App* you can modify the order in which models are displayed simply dragging and dropping them (thanks to twentytab-sortable)

change the model's icon image (only if you selected one between *Use apps and models icon system*, *Use models' icons system in index group models by app* or *Use models' icons system in index ungroup models by app* in **View Mode** in your *Custom Admin* configuration).

In the end, differently from *Custom App*, if **Autocompelte Model** field in your *Custom Admin* configuration is unchecked then only models that you created here are shown.

4.4 Custom Link

`twentytab-customadmin` allows you to create custom links (asbolute or relative) that will be shown in the admin homepage with your models.

Like the apps and the models you can give them an image (if no image is provided then the default app icon will be displayed).