
Picturosaure

Release 0.1

Oct 13, 2020

Contents:

1	Installation	3
1.1	Database	3
1.2	Clone and install dependencies	3
1.3	Apply migrations and create superuser	3
1.4	Development installation	4
2	Configuration	5
2.1	Django settings	5
2.2	Picturosaure settings	6
3	Use Picturosaure	9
3.1	Icons	9
3.2	Licenses	10
3.3	Pictures	10
	Index	13

Picturosaure is a small web application to share pictures.
This is documentation of version 0.1.

1.1 Database

You will need a working database for this project. We recommend the use of PostgreSQL but any database working with django (see [here](#)) will do the trick.

However, please note that this software should be linked to mail server softwares like postfix and dovecot and that any database might not work with those. Please see the documentations of those before choosing any database.

You will need to create, before installation, a database (e.g. `dinomail`) and a user (e.g. `dinomail`) with some password that we will denote `secret` for the rest of this page.

1.2 Clone and install dependencies

First you will have to clone the Github repository of the project. We recommend you to clone from the last release.

```
git clone https://github.com/nanoy42/dinomail
```

Then you need to install the dependencies. There is a Pipfile, from which you can just do

```
pipenv install
```

Or you can use the `requirements.txt` file :

```
pip3 install -r requirements.txt
```

1.3 Apply migrations and create superuser

You can populate the database with the schema with the command

```
python3 manage.py migrate
```

You can then create a superuser by running the command

```
python3 manage.py createsuperuser
```

Warning: You may have to manually create directories `src/media/original` and `src/media/watermark` and give the right permissions.

1.4 Development installation

You can install the dev requirements with

```
pipenv install --dev
```

or

```
pip3 install -r dev-requirements.txt
```

Warning: You may have to manually create directories `src/media/original` and `src/media/watermark` in order to successfully run the tests.

There are two phases of configuration : the first one is related to django and the second one is specific to Picturosaure. In the `src/picturosaure` folder there is file `local_settings.example.py`. Copy it in the same directory as `local_settings.py` :

```
cp src/picturosaure/local_settings.example.py src/picturosaure/local_settings.example.  
→py
```

2.1 Django settings

Please see the [django documentation](#) for extended documentation.

SECRET_KEY

Default: ""

A secret key for a particular Django installation. This is used to provide cryptographic signing, and should be set to a unique, unpredictable value. This value should be kept secret.

Warning: Django can't run without a secret key.

DEBUG

Default: False

A boolean that turns on/off debug mode. You should use `DEBUG=False` for production.

ALLOWED_HOSTS

Default: []

A list of strings representing the host/domain names that this Django site can serve.

Warning: Django can't run with `DEBUG=False` and `ALLOWED_HOSTS=[]`.

DATABASES**Default:** “{

```
    “default”: { “ENGINE”: “django.db.backends.sqlite3”, “NAME”: BASE_DIR / “db.sqlite3”,
    }
```

}“

If you use a postgresql database, on the same host as where you installed DinoMail, with the above values, it should look like this:

```
DATABASES = {
    "default": {
        "ENGINE": "django.db.backends.postgresql",
        "NAME": "dinomail",
        "USER": "dinomail",
        "PASSWORD": "secret",
        "HOST": "localhost",
    }
}
```

LANGUAGE_CODE**Default:** "en-us"

User will not be able to change the interface language. However, you can select the language you want from the listed below :

- English ('en-us')
- French ('fr')

Those are the languages currently supported for DinoMail.

TIME_ZONE**Default:** "UTC"

Time zone of the server.

STATIC_ROOT**Default:** `BASE_DIR / "staticfiles"`

Folder in which the static files should be copied. You should make an alias for `/static` to this directory.

MEDIA_ROOT**Default:** `BASE_DIR / "media"`

Folder in which the media files should be uploaded. You should make an alias for `/media` to this directory.

2.2 Picturosaure settings

There are some settings for picturosaure :

Default : `True`

If set to False, no watermark is added when the “add watermark” function is executed.

Default : "bottom-right "

Position of the watermark. Should be one of the following :

- "top-left "
- "top-right "
- "bottom-left "
- "bottom-right "

PICTUROSAURE_WATERMARK_MARGIN

Default: 20

Margin to apply from the selected watermark position.

PICTUROSAURE_WATERMARK_COLOR

Default : (255, 255, 255, 128)

Color of the watermark in RGBA format (0-255).

PICTUROSAURE_WATERMARK_SIZE

Default: 100

Size of the watermark.

PICTUROSAURE_WATERMARK_TEXT

Default: "©"

Text to print in the watermark.

PICTUROSAURE_TITLE

Default: "Picturosaure"

Text displayed in title.

PICTUROSAURE_CONTACT

Default: "example@example.org"

Contact email. If set to “”, the mail icon is not displayed.

PICTUROSAURE_LICENSE

Default: `""Where not stated otherwise, the photograph is distributed under the CC BY-NC-SA 4.0 license (attribution to).""`

The text is displayed on the right panel. Is supposed to give the default license of the pictures.

PICTUROSAURE_LICENSE_NAME

Default: ""

Text to display right to the HTML5UP license information. If set to "" nothing is displayed.

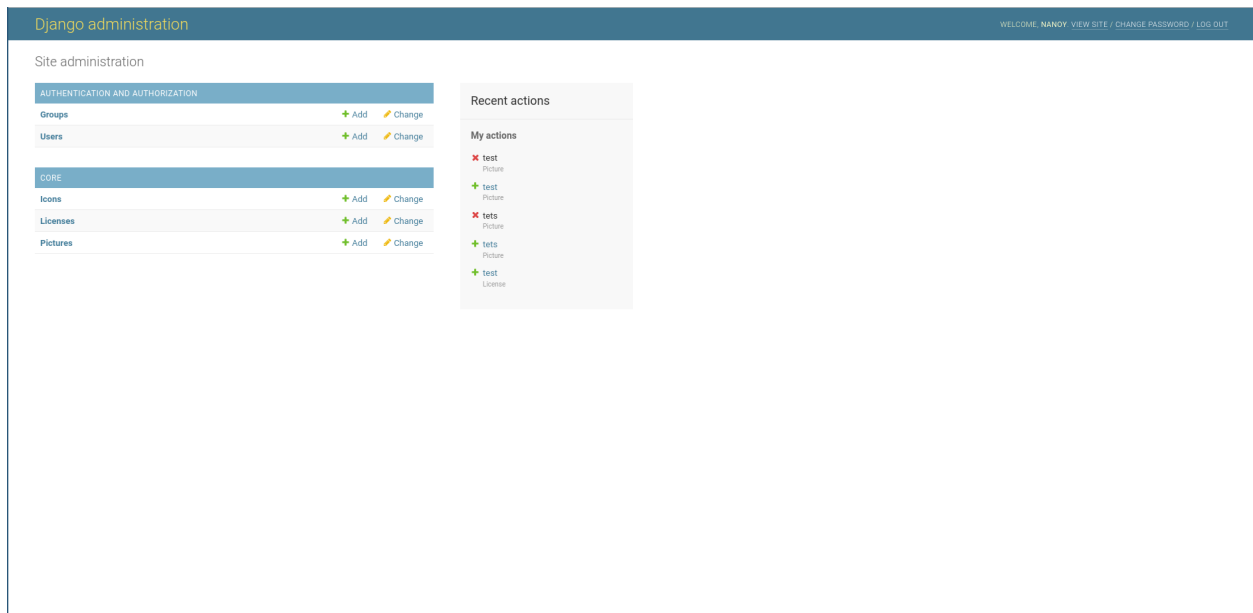
CHAPTER 3

Use Picturosaure

Picturosaure is composed of a unique web page that display pictures. All the admin work is done in the django admin section.

If picturosaure runs at `pictures.example.org` you can access the admin panel at `pictures.example.org/admin`.

You will need an admin account, see installation for how to create a superuser account.



3.1 Icons

Icons are displayed in the right panel.

An icon is composed of a name (which is not not displayed if everything is correct), a link and a font-awesome icon. The font awesome icon should be the whole class, for example : `fab fa-instagram`.

Note: The mail icon is not handled here as django doesn't recognize it as a URL. It is handled through the `PICTUROSAURE_CONTACT` attribute.

3.2 Licenses

Pictures will have a corresponding license. If all your pictures are distributed under the same license, you will need to create the instance only once.

A license is just a name and a link to the full license text.

3.3 Pictures

3.3.1 Add pictures

When adding pictures, the following fields are mandatory:

- name : the name of the pictures
- picture : the image
- license : the license object

Optionally, you can add :

- a small description
- a date
- a location

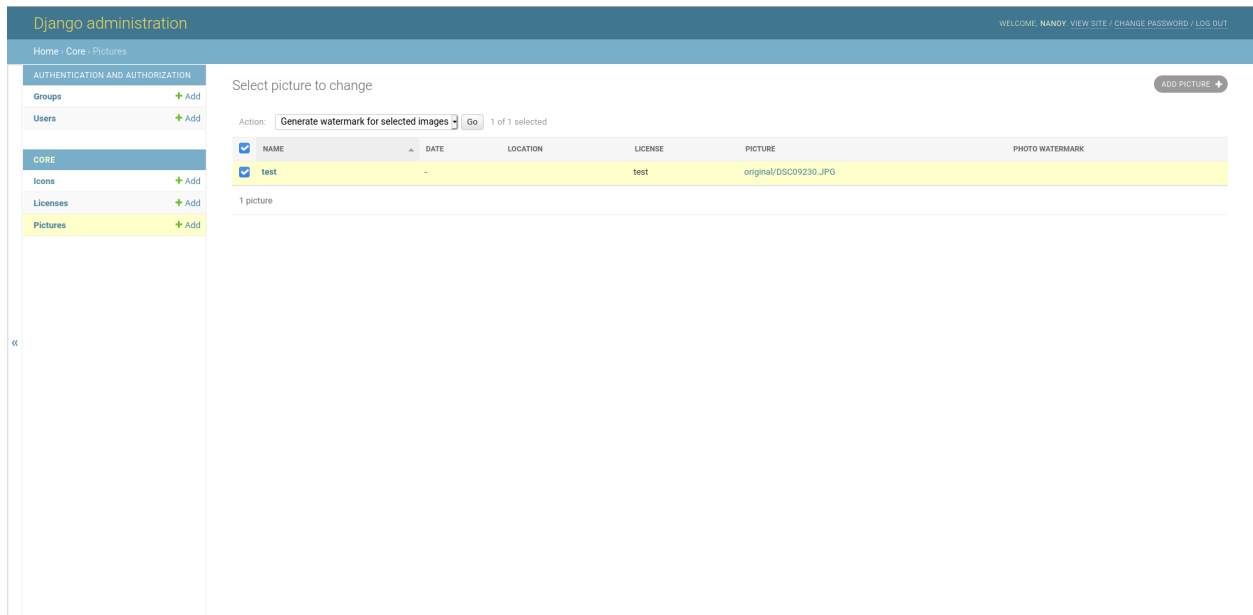
3.3.2 Add watermarks

Warning: Adding watermarks is mandatory to display the picture on the front page. Otherwise it will just appear as black picture. If you don't want any watermark, please set `PICTUROSAURE_USE_WATERMARK` to `False` and generate the watermarks.

To generate watermarks, you have to select the corresponding pictures on the admin page and select the action 'Generate watermark for selected images'.

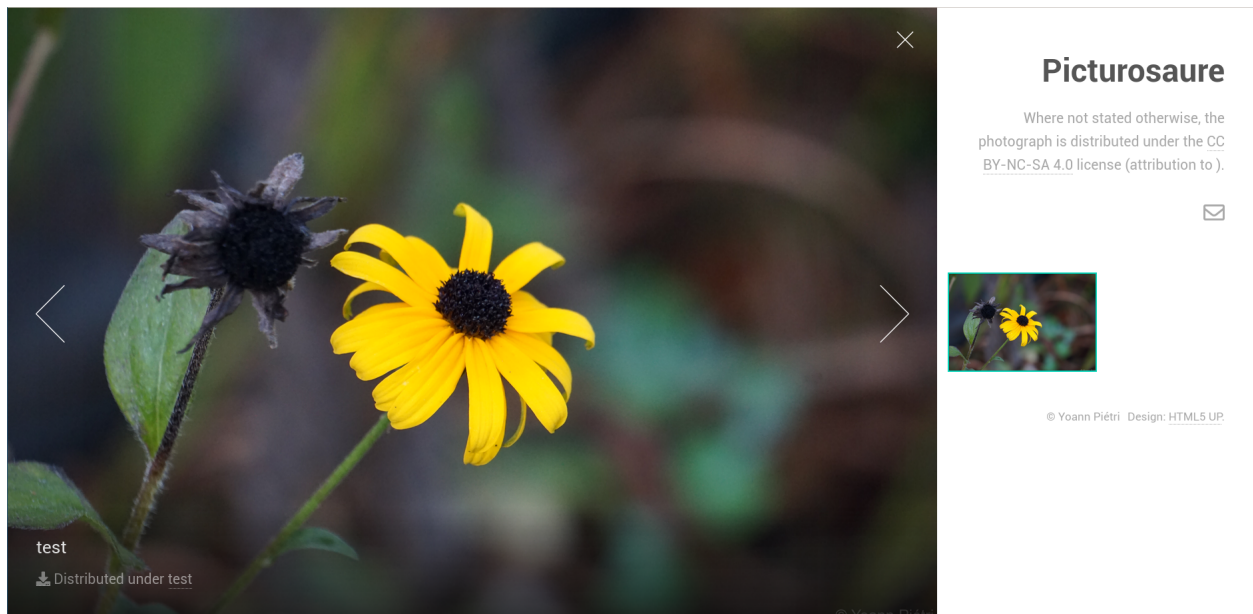
This operation can take a while depending on the picture's size.

If a selected pictures has already a watermark, it is deleted and a new one is generated.



3.3.3 Remove pictures

If you remove pictures, the corresponding files are automatically deleted on the server.



A

ALLOWED_HOSTS, 5

D

DATABASES, 6

DEBUG, 5

L

LANGUAGE_CODE, 6

M

MEDIA_ROOT, 6

P

PICTUROSAURE_CONTACT, 7

PICTUROSAURE_LICENSE, 7

PICTUROSAURE_LICENSE_NAME, 7

PICTUROSAURE_TITLE, 7

PICTUROSAURE_WATERMARK_COLOR, 7

PICTUROSAURE_WATERMARK_MARGIN, 7

PICTUROSAURE_WATERMARK_SIZE, 7

PICTUROSAURE_WATERMARK_TEXT, 7

S

SECRET_KEY, 5

STATIC_ROOT, 6

T

TIME_ZONE, 6