
Prosopopee Documentation

Release latest

May 10, 2022

Contents

1	Why	3
2	Licence	5
3	Credit	7
4	Documentation	9
4.1	Installation	9
4.2	Configuration	10
4.3	Sections	17
4.4	Example	22
4.5	Build the website	24
4.6	Theming	25
4.7	Authors	26
4.8	Changelog	26

More or less a small clone of exposure.co in form of a static website generator. For those of you who don't know what exposure.co is, it allows you to tell a story with your pictures.

You can find example usages here:

- <http://surlchemesins.fr>
- <http://media.faimaison.net/photos/galerie/>
- <https://www.thebrownianmovement.org/>
- <http://outside.browny.pink>
- <http://such.life>
- <http://www.street-art.me>

CHAPTER 1

Why

I wanted to learn a bit of advanced css and I wanted to self-host my data instead of using exposure.co.

CHAPTER 2

Licence

GPLv3+

CHAPTER 3

Credit

```
16:57 <meornithorynque> et tu as besoin d'un nom ?  
16:57 <meornithorynque> genre n'importe quoi ?  
16:57 <meornithorynque> je propose "Prosopopée"
```


4.1 Installation

4.1.1 Requirements

Ubuntu/Debian

We need Python, pip and virtualenv:

```
apt-get install python3-pip python3-virtualenv
```

and graphicsmagick library for building the gallery:

```
# graphicsmagick requires to have the 5.3.1 version of gcc-5-base  
apt-get install graphicsmagick
```

A video converter like ffmpeg:

```
apt-get install ffmpeg
```

or:

```
apt-get install libav-tools
```

For deployment, we need rsync:

```
apt-get install rsync
```

Mac

We need Brew:

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/  
↪master/install)"
```

and graphicsmagick library for building the gallery:

```
brew install graphicsmagick
```

A video converter like ffmpeg:

```
brew install ffmpeg
```

For deployment, we need rsync:

```
brew install rsync
```

4.1.2 Installation in virtualenv

1. Create a virtualenv, and activate it:

```
virtualenv ve  
source ve/bin/activate
```

2. Download and install Prosopopee:

```
pip3 install prosopopee
```

4.1.3 Docker

Get the Docker image:

```
docker pull beudbeud/prosopopee
```

Run:

```
docker run --rm -v $(pwd):/site beudbeud/prosopopee
```

More informations <https://hub.docker.com/r/beudbeud/prosopopee/>

4.2 Configuration

The files organisation is quite simple:

- in the root directory of your project you need a settings.yaml file that contains the title and subtitle of your gallery
- for each gallery you need a folder that also contains a settings.yaml file that describes how to display the content in your gallery
- and you put the pictures of the gallery inside the gallery folder
- or other directory, in the gallery became an index (so pictures won't be display, only cover of child gallery)

4.2.1 Root settings.yaml

The root settings.yaml should contain 2 keys: one for the title of your website and one for the subtitle. It should look like that:

```
title: My exploration of the outside world
sub_title: it's a scary place, don't go there
```

It can also optionally contain a menu and global settings.

Menu

It is possible to add a menu to your homepage that links to static pages. To do so, add a *menu* key to your *settings.yaml*, for example:

```
title: "About"
menu:
  - about: "About"
  - first_gallery: "My first gallery"
  - http://twitter.com: "Twitter"
```

For example, this could be the content of *settings.yaml* in *about* folder:

```
title: "About"
static: true
public: false
sections:
  - type: text
    text: Some text, HTML <b>is allowed</b>.
```

You can use the *static* option to get a template closer to the one of the homepage that is better suited for a static page. You'll need to specify "public: false" if you don't want to list this page on the homepage. If you don't specify "public: false" you **need** to specify a "cover:" entry like any other gallery.

NOTE: expect the "static: " option to disappear quite soon for a more generic approach to "choose your page style".

Include Files

You can include files with the *include* option. These files will be copied as is from the root of your project directory to the build directory. Example:

```
title: My photos
sub_title: wow
include:
  - robots.txt
  - .well-known/brave-rewards-verification.txt
```

Global settings

Global settings can be put in your root *settings.yaml*, under the *settings* key.

GM

Currently a *gm* setting key allows to customize the default GraphicsMagick's behaviour. It looks like:

```
title: Gallery
settings:
  gm:
    quality: 75
    auto-orient: True
    strip: True
    resize: 50%
    progressive: True
```

The meaning of the currently supported GraphicsMagick's settings is as follows:

- *quality* allows to customize the compression level of thumbnails (between 0 and 100)
- *auto-orient* changes the orientation of pictures so they are upright (based on corresponding EXIF tags if present)
- *strip* removes all profiles and text attributes from the image (good for privacy, slightly reduce file size)
- *resize* can be used to resize the full-size version of pictures. By default, input image size is preserved
- *progressive* converts classic baseline JPEG files to progressive JPEG, and interlaces PNG/GIF files (improves the page loading impression, slightly reduces file size)

Any GraphicsMagick setting can be customized on a per-image basis (either *cover* or *image*, see below).

Video converter

Prosopopée can use ffmpeg or libav and each can be configured if needed:

```
title: Gallery
settings:
  ffmpeg:
    binary: "ffmpeg"
    loglevel: "error"
    format: "webm"
    resolution: "1280x720"
    vbitrate: "3900k"
    abitrage: "100k"
    audio: "libvorbis"
    video: "libvpx"
    other: "-qmin 10 -qmax 42 -maxrate 500k -bufsize 1500k"
```

The meaning of the currently supported FFMEG or LIBAV's settings is as follows :

- *binary* sets the binary to use to convert the video (ffmpeg or avconv)
- *loglevel* sets the logging level used by the library
- *format* forces input or output file format
- *resolution* sets frame size
- *vbitrate* sets video bitrate
- *abitrage* sets audio bitrate
- *audio* sets the audio codec
- *video* sets the video codec

- *extension* sets the extension of output file
- *other* sets different options if you need more

example for MP4:

```
title: Gallery
settings:
  ffmpeg:
    binary: "ffmpeg"
    format: "mp4"
    audio: "acc"
    video: "libx264"
    extension: mp4
```

Light Mode

To enable the light mode:

```
title: Gallery
settings:
  light_mode: true
```

With this option Prosopopee makes a subdirectory with light version of your gallery. This light gallery uses less JS, pictures in low size, etc.

To access this light gallery, add /light at the end of the URL of the gallery.

If you want only light theme, see themes.

Night Mode

To enable the night mode, which is only available for exposure theme (default theme):

```
title: Gallery
settings:
  night_mode: true
```

After that you will be able to choose between dark theme or light theme when visiting the website.

Themes

Prosopopée supports different themes. As for now, only 3 themes are available:

- the default one called “exposure”
- “material” based on materialcss
- light

To specify the theme, add the “theme” key in your “settings” key or your **root** settings.yaml. For example:

```
title: My exploration of the outside world
sub_title: it's a scary place, don't go there
settings:
  theme: material
```

Licence

By default Prosopopée uses CC-BY-SA for all the content. If you want to use another licence, you need to add a “licence” key in **root** settings.yaml. For example:

```
title: Gallery
licence:
  name: WTFPL
  url: "http://www.wtfpl.net/txt/copying/"
```

Share

If you want to enable content sharing on social networks, add the “share” key in **root** settings.yaml. For example:

```
title: Gallery
share: true
url: "http://prosopopee.com"
```

By default you can share on facebook, twitter, pinterest, google+.

RSS

To activate the RSS feed you need to add the “rss” key in **root** settings.yaml:

```
title: Gallery
rss: true
url: "http://prosopopee.com"
```

Open Graph Meta

To activate the Open Graph Meta you need to add the “og” key in **root** settings.yaml:

```
title: Sur les chemins
url: "http://prosopopee.com"
settings:
  og: true
```

You can also specify a description and a language (“lang” key) for a gallery.

For more information about Open Graph <http://ogp.me/>

Deployment

If you want to configure the deployment of your website with rsync:

```
title: Gallery
settings:
  deploy:
    ssh: true (optional, for ssh)
    username: username (optional, for ssh)
    hostname: server.com (optional, for ssh)
```

(continues on next page)

(continued from previous page)

```
dest: /var/www/website/build/  
others: --delete-after (optional)
```

Reverse order

Normally Prosopopee builds the gallery index in anti-chronological. If you want to have it in chronological order, use the “reverse” key:

```
settings:  
  reverse: true
```

This option can also be used in a gallery settings.yaml if you use multi-level galleries:

```
title: Multi level gallery  
reverse: true
```

Password access

If you want to protect all the website by password:

```
title: Gallery  
password: my_super_password
```

Date locale

By default, Prosopopee uses locale from LC_TIME environment variable to generate human-readable date.

For example, if your LC_TIME equals to en_US.utf8, then you get “23 August 2020” as date on the gallery tile. If you are using ru_RU.utf8, then you get “23 2020”.

If you want to use non-based on LC_TIME locale for human-readable dates on gallery, use the “date_locale” key:

```
settings:  
  date_locale: ru_RU
```

4.2.2 Gallery settings.yaml

This settings.yaml will describe:

- the title, subtitle and cover picture of your gallery that will be used on the homepage
- the tags, which are optional
- if your gallery is public (if not, it will still be built but won’t appear on the homepage)
- the date of your gallery that will be used on the homepage since **galleries are sorted anti-chronologically**
- the list of sections that represents your gallery. A section represents either one picture, a group of pictures or text. The different kinds of sections will be explained in the next README section.

Example

```
title: Gallery title
sub_title: Gallery sub-title
date: 2016-01-15
cover: my_cover_picture.jpg
description: Some text
lang: en_US
tags:
  - #yolo
  - #travel
sections:
  - type: full-picture
    image: big_picture.jpg
    text:
      title: Big picture title
      sub_title: Some text
      date: 2016-01-15
  - type: pictures-group
    images:
      -
        - image1.jpg
        - image2.jpg
        - image3.jpg
      -
        - image4.jpg
        - image5.jpg
  - type: text
    text: Some text, HTML <b>is allowed</b>.
  - type: bordered-picture
    image: another_picture.jpg
```

And here is an example of a **private** gallery (notice the `public` keyword):

```
title: Gallery title
sub_title: Gallery sub-title
date: 2016-01-15
cover: my_cover_picture.jpg
public: false
sections:
  - ...
```

4.2.3 Advanced settings

Image handling

Images go into the `cover` or `image` keys. Each image individual processing settings can be customized to override the default GraphicsMagick settings defined (or not) in the root `settings.yaml`.

This is done by putting the image path into a `name` key, and adding specific processing settings afterwards.

For example, you can replace:

```
image: image1.jpg
```

by:

```
image:
  name: image1.jpg
  quality: 90
  strip: False
  auto-orient: False
```

Password access

You can restrict access to a gallery with a password:

```
title: Gallery title
sub_title: Gallery sub-title
password: my_super_password
```

4.3 Sections

A gallery is composed of a succession of sections as you can see on this [wonderfully totally uninteresting example gallery](#). This gallery is split in 5 sections:

- a full screen picture with text written on it
- a picture with borders around it
- a group of 5 pictures
- and a full-screen picture without text on it this time

In your settings.yaml, a section will **always** have a `type` key that will describe its kind and additional data. Underneath, the `type` key is actually the name of an HTML template and the other data will be passed to this template.

You can find all the section templates here:

<https://github.com/Psycojoker/prosopopee/tree/master/prosopopee/themes/exposure/templates/sections>

You often have an `image` key. You need to give it a path to the actual file. By convention, those files are put inside your gallery folder but this is not mandatory.

4.3.1 Full Screen picture

This displays a full screen picture as shown in the [example gallery](#) in the first and last sections. How you should use it:

With text:

```
- type: full-picture
  image: big_picture.jpg
  text:
    title: Big picture title
    sub_title: Some text
    date: 2016-01-15
    date_end: 2016-01-24 (Optional)
```

Without text:

```
- type: full-picture
  image: big_picture.jpg
```

If you want a fixed background you can use this option (only with the exposure theme):

```
- type: full-picture
  fixed: true
```

The *background_position* setting can be used to control how the image is centered when it doesn't fit the screen width. This is useful for controlling the center across different screen sizes (e.g. mobile), keeping the focal element of an image always visible. This sets the *background-position* css property. This feature is supported by the "exposure" theme. Example:

```
- type: full-picture
  image: big_picture.jpg
  background_position: 75%
```

4.3.2 Bordered picture

This displays a centred picture that is surrounded by white (the background) as shown in the second position of the [example gallery](#).

How to use it:

```
- type: bordered-picture
  image: another_picture.jpg
```

4.3.3 Group of pictures

This displays a group of zoomable pictures on one or multiple lines as shown on the fourth position (after the text) of the [example gallery](#):

```
- type: pictures-group
  images:
    -
      - image1.jpg
      - image2.jpg
      - image3.jpg
    -
      - image4.jpg
      - image5.jpg
```

The first level - represents a row of pictures. The second level - represents the list of images in this line.

Known bug: the images are left aligned, so if you don't put enough images on a row, you'll have some white space on the right.

4.3.4 Text

This displays some centred text as shown on the third position of the [example gallery](#). HTML is allowed inside the text.

How to use it:

```
- type: text
  text: Some text, HTML <b>is allowed</b>.
```

4.3.5 Paragraph

This displays a h2 title followed by text. HTML is allowed inside the text. If no title is declared, a separator is added.

How to use it:

```
- type: paragraph
  title: the title
  text: Some text, HTML <b>is allowed</b>.
```

Since version 0.5 you can add a floating image in the paragraph:

```
- type: paragraph
  title: the title
  text: Some text, HTML <b>is allowed</b>.
  image:
    name: image.jpg
    float: right
    size: 150px
```

By default if you don't set float and size the image will be on left with a size of 250px.

4.3.6 HTML

This section is for raw html that will be centred (for example: inlining an OSM iframe).

How to use it:

```
- type: html
  html: <tag>some html stuff</html>
```

4.3.7 Panorama

This displays a very large picture that can be drag-and-dropped.

How to use it:

```
- type: panorama
  image: 7.jpg
```

4.3.8 Audio

This section is for adding an audio file playable with the HTML5 player.:

```
- type: audio
  title: Title of song
  image: song.ogg
  color: "#000" (optional)
```

4.3.9 Author

This section is for describing the author of the story:

```
- type: author
  name: Adrien Beudin
  text: Some text
  image: IMG_20150725_200941.jpg
  twitter: beudbeud (Optional)
  facebook: beudbeud (Optional)
  website: plop.fr (Optional)
```

4.3.10 Iframe

This section makes your embed iframes responsive:

```
- type: iframe
  name: <iframe width="560" height="315" src="https://www.youtube.com/embed/
↵nshFXWEKxs4" frameborder="0" allowfullscreen></iframe>
```

4.3.11 Quote

To use quote blocks easily:

```
- type: quote
  text: This is a quote
```

4.3.12 Advanced options

Image caption

Prosopopée supports captions for images, you can use it on bordered-picture and pictures-group.

Example on bordered-picture:

```
- type: bordered-picture
  image: another_picture.jpg
  text: This is a caption
```

And on pictures-group:

```
- type: pictures-group
  images:
    -
      - name: image1.jpg
        text: This is a caption
      - image2.jpg
      - image3.jpg
    -
      - image4.jpg
      - image5.jpg
```


Background settings

For all sections, you can define the background.

Example for background colour:

```
- type: bordered-picture
  background: "#555"
  image: another_picture.jpg
```

or you can use a picture:

```
- type: text
  background: "url(background_picture.jpg) "
  text: Some text
```

Text color settings

For text, html and paragraph sections, you can also define the text colour.

Example:

```
- type: bordered-picture
  color: "#333"
```

Video support

For bordered-picture, full-picture and pictures-group, it's possible to use video instead of pictures. You have to specify with the "type" key that it's a video.

The video will be converted using either ffmpeg or avconv (depending on the one specified in the settings, ffmpeg being the default one).

Example for pictures-group:

```
- type: pictures-group
  images:
  -
    - name: video.mp4
      type: video
    - image1.jpeg
    - image2.jpeg
  -
    - image3.jpeg
    - image4.jpeg
```

Example for bordered-picture:

```
- type: bordered-picture
  image:
    name: video.mp4
    type: video
```

And for full-picture:

```
- type: full-picture
  image:
    name: video.mp4
    type: video
  text:
    title: Title Text
    sub_title: Sub title text
    date: 2016-03-11
    date_end: 2016-03-25
```

If you want enable the controls:

```
- type: bordered-picture
  image:
    name: video.mp4
    type: video
  controls: true
```

You can also use a video as a gallery cover:

```
title: pouet
sub_title: plop
cover:
  name: video.mp4
  type: video
```

4.4 Example

As a recap, here is how the files of the example gallery are organised:

```
example
├── settings.yaml
├── first_gallery
│   ├── settings.yaml
│   └── stuff.png
```

The content of `example/settings.yaml`:

```
title: "Example gallery"
sub_title: some subtitle
```

The content of `example/first_gallery/settings.yaml`:

```
title: my first gallery
sub_title: some subtitle
date: 2015-12-08
cover: stuff.png
sections:
- type: full-picture
  image: stuff.png
  text:
    title: Beautiful Title
    sub_title: pouet pouet
    date: 2015-12-08
- type: bordered-picture
```

(continues on next page)

(continued from previous page)

```

image: stuff.png
- type: text
  text: « voici plein de blabla à rajouter et <b>ceci est du gras</b> et encore
↳plein plein plein plein de text car je veux voir comment ça va wrapper car c'est
↳important et il faut pas que j'oublie de mettre des margins en % sinon ça va pas le
↳faire alala là ça devrait aller »
- type: pictures-group
  images:
  -
    - stuff.png
    - stuff.png
    - stuff.png
  -
    - stuff.png
    - stuff.png
- type: full-picture
  image: stuff.png

```

The content of `example/second_gallery/settings.yaml`:

```

title: my second level gallery
sub_title: some subtitle
date: 2015-12-08
cover: stuff.png

```

The content of `example/second_gallery/second_level_gallery_2/settings.yaml`:

```

title: my second level gallery 2
sub_title: some subtitle
date: 2015-12-08
cover: stuff.png
sections:
- type: full-picture
  image: stuff.png
  text:
    title: Beautiful Title
    sub_title: pouet pouet
    date: 2015-12-08
- type: full-picture
  image:
    name: video.mp4
    type: video
  text:
    title: Beautiful Title
    sub_title: pouet pouet
    date: 2015-12-08
- type: bordered-picture
  image:
    name: video.mp4
    type: video
- type: bordered-picture
  image: stuff.png
- type: bordered-picture
  image:
    name: video.mp4
    type: video
  text: "plop"

```

(continues on next page)

(continued from previous page)

```
- type: bordered-picture
  image: stuff.png
  text: "plop"
- type: text
  text: « voici plein de blabla à rajouter et <b>ceci est du gras</b> et encore_
↳plein plein plein plein de text car je veux voir comment ça va wrapper car c'est_
↳important et il faut pas que j'oublie de mettre des margins en % sinon ça va pas le_
↳faire alala là ça devrait aller »
- type: pictures-group
  images:
  -
    - name: stuff.png
      text: "test"
    - name: video.mp4
      type: video
    - name: stuff.png
      text: "test"
  -
    - stuff.png
    - name: video.mp4
      type: video
- type: pictures-group
  images:
  -
    - name: stuff.png
      text: "test"
    - name: stuff.png
      text: "test"
    - name: stuff.png
      text: "test"
  -
    - stuff.png
    - stuff.png
- type: full-picture
  image: stuff.png
- type: full-picture
  image:
    name: video.mp4
    type: video
```

4.5 Build the website

For build your website there are two way:

- Auto generation
- Self made

If you want do all by your self and make custom gallery and use many section, you need follow <https://prosopopee.readthedocs.io/en/latest/configuration.html#gallery-settings-yaml>

But if is just for make gallery with only picture you can generate it automatically.

You need

1. Create a folder

2. Put all pics you want
3. Create settings.yaml file in the folder
4. Add title, date and cover key in folder/settings.yaml
5. Use *prosopopee autogen -d folder*

4.5.1 Generate

Note: You need to be in an activated virtualenv.

In a folder containing the **root** settings.yaml file, simply do:

```
prosopopee
```

A *build* folder will be created in the current directory, containing an index.html, static files (css & js) and pictures.

4.5.2 Preview

In a root folder launch this command:

```
prosopopee preview
```

Then, you can check your website at <http://localhost:9000>

You can also provide custom port for the webserver:

```
prosopopee preview -p 8080
```

And proceed to <http://localhost:8080>

4.5.3 Deployment

Prosopopee can upload your website with rsync, to do so, run:

```
prosopopee deploy
```

4.6 Theming

4.6.1 Custom css and js

You can easily add css code and js code. You just need to create a custom.css or custom.js in root directory.

4.6.2 Override template

If you want to override a template, you need to create a “templates” directory and add your own template.

- home page : index.html
- gallery page: gallery-index.html

And to override sections you need to create a file in “templates/sections”.

4.6.3 Create theme

TODO

4.7 Authors

By chronological order:

- Bram, launched the project
- Kload
- opi
- taziden
- beudbeud
- CapsLock
- Julien Malik
- Titoko
- 0x010C
- QSchulz

4.8 Changelog

1.1.4 (2020-11-04)

- Make Exposure theme work better on mobile by toki

1.1.3 (2020-10-21)

- JQuery fixes for issues #121 and #126 by QSchulz

1.1.2 (2020-10-13)

- fix weird upload bug on pypi

1.1.1 (2020-10-13)

- upgrade baguetteBox.js to 1.11.1 for its bug fix

1.1 (2020-10-12)

- move to tox with check-manifest, flake8 and black
- move to the logging module
- move to argparse
- localise date formatting <https://prosopopee.readthedocs.io/en/latest/configuration.html#date-locale>
- allow to customize background-position for full-picture <https://prosopopee.readthedocs.io/en/latest/sections.html#full-screen-picture>
- add include option <https://prosopopee.readthedocs.io/en/latest/configuration.html#include-files>
- various bug fixes

Thx to all new contributors!

1.0.1 (2020-05-08)

- Fix check date format
- Fix error datetime
- lazyload for video
- add controls on fullscreen video

1.0.0 the covid release (2020-04-29)

- Add automatic generation gallery
- Little refactoring

0.9.2

- Catch traceback when date format is in incorrect format

0.9.1

- update pypi package
- Recommended to use python 3

0.9.0

- Fix audio player on mobile
- Update JS lib
- Change lazyload lib
- Possibility the add controls on video
- Update exposure theme
- fix some issue

0.8.3 (2020-02-24)

- Fix pip package

0.8.2 (2019-09-25)

- Fix encryption page
- Fix URI local (Thanks QSchulz)

0.8.1 (2018-03-28)

- Fix some css error
- Update Material theme
- Add fadeInUp effect in exposure theme

0.8 (2018-02-28)

- Add night mode https://prosopopee.readthedocs.io/en/latest/configuration.html#night_mode
- Add a test command to build html without generating media files
- fix some bugs

0.7 (2017-10-04)

- Add password access (based on <https://robinmoisson.github.io/staticcrypt/>)

0.6 (2017-07-14)

- Compatibility python 2 and 3
- Possibility to add custom css and js <http://prosopopee.readthedocs.io/en/latest/theming.html>
- Add reverse option by titoko <https://prosopopee.readthedocs.io/en/latest/configuration.html#reverse>

0.5 (2017-06-04)

- Add audio HTML5 player <https://prosopopee.readthedocs.org/en/latest/sections.html#audio> by beudbeud
- Update Material theme by beudbeud
- Add iframe section <https://prosopopee.readthedocs.org/en/latest/sections.html#iframe> by beudbeud
- Add quotes section <https://prosopopee.readthedocs.org/en/latest/sections.html#quote> by beudbeud
- Add deploy and preview option <https://prosopopee.readthedocs.io/en/latest/build.html#preview> by beudbeud
- Load only css and jss if the section is used by beudbeud
- Possibility to add floating image in paragraph by beudbeud <https://prosopopee.readthedocs.org/en/latest/sections.html#paragraph>
- fix some bugs

0.4 (2016-12-11)

- greatly improved loading speed of pages with several different technics (see below)
- RSS <https://prosopopee.readthedocs.org/en/latest/configuration.html#rss> by beudbeud
- possibility to use video in section and cover by beudbeud (and a bit of Bram)
- add lazyload for pics by beudbeud
- if a theme doesn't have a section, fallback on exposure theme which is considered the default one by titoko
- code and templates cleaning by Bram
- make code a bit more robust by Bram
- basic CI on travis by Bram
- Light mode by beudbeud
- progressive JPEG/GIF/PNG by default for a better loading experience by 0x010C following sebian's blogpost
- <picture> element support for smoother loading by Bram
- resposive mode of baguette by Bram
- several background images for smoother loading by Bram
- optimise write time to avoid blank pages during regeneration by Bram
- optional.opengraph support by beudbeud <https://prosopopee.readthedocs.io/en/latest/configuration.html#open-graph-meta>

0.3.1 (2016-04-13)

- fix: cover date was hidden by default, this was a backward breaking behavior

0.3 the "beudbeud release" (2016-04-13)

- caption support on bordered picture and pictures group <https://prosopopee.readthedocs.org/en/latest/sections.html#advanced-options> by beudbeud
- configure licence <https://prosopopee.readthedocs.org/en/latest/configuration.html#licence> in footer by beudbeud

- possibility to use a range for the full picture date <https://prosopopee.readthedocs.org/en/latest/sections.html#full-screen-picture> by beudbeud
- Update material theme by beudbeud
- social share <https://prosopopee.readthedocs.org/en/latest/configuration.html#share> by beudbeud
- Define background and text color of section <https://prosopopee.readthedocs.org/en/latest/sections.html#advanced-options> by beudbeud

0.2 (2016-02-23)

- a lot new contributors stepped in, see <https://prosopopee.readthedocs.org/en/latest/authors.html>
- possibility to specify an (optional) menu <https://prosopopee.readthedocs.org/en/latest/configuration.html#menu> made by beudbeud
- configure Graphics Magick options on a global or per images fashion <https://prosopopee.readthedocs.org/en/latest/configuration.html#gm> and <https://prosopopee.readthedocs.org/en/latest/configuration.html#images-handling> by capslock and jmk
- support for various themes and a new material theme in addition of the default one <https://prosopopee.readthedocs.org/en/latest/configuration.html#themes> by beudbeud
- tags support in gallery settings <https://prosopopee.readthedocs.org/en/latest/configuration.html#example> by beudbeud
- introduce internal CACHE format version to avoid breaking build when changing it

0.1 (2016-02-09)

- First pypi release