



MySmile Documentation

Release 0.7.1

MySmile team

October 15, 2015

Contents

1. Getting started	3
1.1. Overview	3
1.2. Quick install guide	3
2. Technical specification	5
2.1. Objective	5
2.2. Features	5
2.3. Technology	6
2.4. Database schema	6
2.5. Urls	7
2.6. Menu	7
2.7. Content	8
2.7.1. Page parts	8
2.7.2. SEO configuration	8
2.7.3. Technical configuration	8
2.7.4. Preferences	9
2.8. Admin panel	9
2.8.1. Overview	9
2.8.2. Language localisation	11
2.9. Useful links	11
3. Themes	13
3.1. Available themes	13
3.2. Configurations of theme	14
3.3. How to add a new theme	14
3.4. How to create a new theme	15
4. REST API	17
4.1. REST API v1	17
4.1.1. Objective	17
4.1.2. Configuration	17
4.1.3. Endpoint	17
4.1.4. Methods	17
4.1.5. General parameters	17
4.1.6. Request	18
4.1.7. Response	18
4.1.8. Resources	18
4.1.9. Errors	20
4.1.10. Examples	21
4.1.11. Useful links	21
5. Frequently Asked Questions	23
6. Glossary	25

Contents:

1. Getting started

1.1. Overview

MySmile is the lightweight open-source CMS based on Django. It helps to create websites with configurable design and minimum cost. MySmile is best choose for:

- Small websites aimed on static content with limited dynamic. For example such: “Startup business”, “Personal site”, “Conference”.
- Some prototypes/ideas that now looks tiny but has enormous potential to grow. Because MySmile use Django framework it breaks all barriers for implementation your new features.
- Freshmen Python developers for studying and experimenting.

Innovation MySmile features:

- Use newspaper text organize technology like: columns, incuts.
- REST API helps construct mobile and desktop application
- Moreover REST API client is available, for now it is php client with examples and unitTests
- Design can be modified by admin panel without any additional program coding.
- There are more then 27 unique designers combination! How we get them? Let’s look into calculation. Every page consists of the 3 parts: central-column, right-column, bottom-column. Right-column can have 3 columns. And right-column has also 3 components: photo, youtube, text. Additionally every element can or can’t be present.
- Additional menu is creating dynamically from pages that is present inside of page as a link. It works like mix of “cloud of links” and “bread crumbs”. So you open links and it’s on the menu.

Source code is available on [GitHub](#).

Get the latest version with the command: `$ git clone --branch=master git@github.com:MySmile/mysmile.git`

See unstable development version: `$ git clone --branch=dev git@github.com:MySmile/mysmile.git`

Please read this documentation or try <http://demo.mysmile.com.ua/> for more information.

1.2. Quick install guide

It is assumed that the [python](#) version 3.x and [bower](#) are already installed on your computer.

1. Download [MySmile](#) and unpacked to the project’s directory
2. Install with one command: `$ make install`
3. In the project directory run a command in the terminal: `$ python3 manage.py runserver`

Now your website is available on <http://127.0.0.1:8000>, link to admin page is <http://127.0.0.1:8000/admin>. Log in with default credentials:

- login: *test*
- password: *test*

2. Technical specification

2.1. Objective

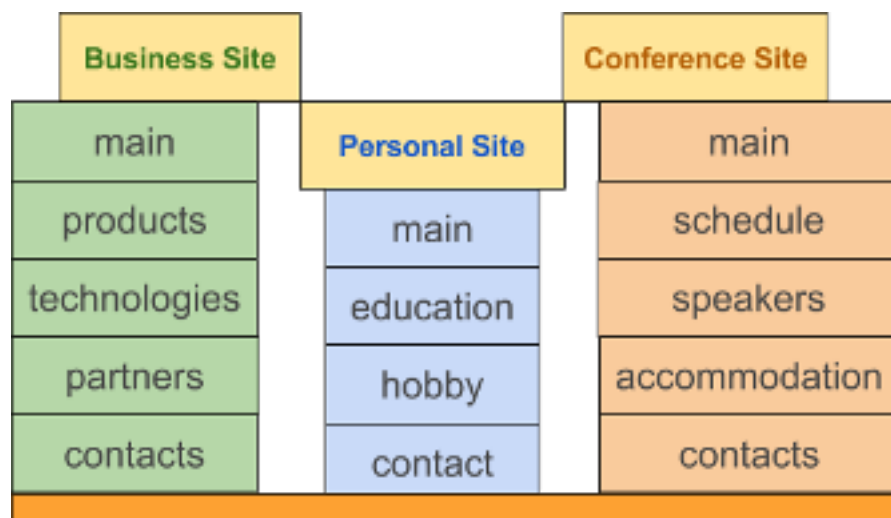
MySmile is the lightweight open-source CMS based on Django. Project focus on context, usability and SEO. It is aimed particularly to small site that provides information about person or business. It is something similar to Blog and Homepage. Information of such site does not change so often as in Blog but more then on Homepage. In other side MySmile can be extended for the way as you want.

MySmile includes:

- Admin panel for content managing
- SEO: friendly url, meta, HTML5 semantic, Google analytics, etc
- Flexible design configuration

Example of site structures that MySmile will be a best choose:

- Small business site with pages: “Main”, “Products”, “Technologies”, “Partners”, “Contacts”.
- Personal site with pages: “Main”, “Education”, “Hobbies”, “Contacts”.
- Conference site with pages: “Main”, “Schedule”, “Speakers”, “Accommodation”, “Contacts”.
- Quick start of Python and Django.



2.2. Features

Fundamental features:

- Internationalization and localization
- Google Analytics is connected
- Sitemap.xml is generated on fly
- Two fully responsive *themes* out of the box
- Switching between the themes from admin
- User interface that “Don’t make me think”

-
- Additional dynamic menu like a mix of “cloud of links” and “bread crumbs”
 - REST API with [REST API Client on PHP, Java, Python](#)
 - Unlimited number of pages

SEO features:

- HTML5 semantic
- Sitemap.xml
- Robots.txt
- Friendly *Urls*
- Meta tags
- Alt attribute for images

See also:

[SEO configuration](#)

2.3. Technology

Project's part	Technology
Language	Python 3.x
Framework	Django 1.8.x
Database	SQLite 3
Frontend	HTML, CSS, JS framework like Bootstrap or Pure CSS
Documentation	Sphinx

Also used third-party components:

1. [JSColor](#) as colorpicker in admin
2. [Bootstrap](#), [Pure CSS](#) into *themes*.

2.4. Database schema

Database schema is designed with a [MySQL Workbench](#). It is available [here](#).



2.5. Urls

URL is user friendly based on unique slug. Valid characters for slug are alpha numeric and hyphen to join words. For instance: <http://mysmile.com.ua/en/about-company.html>

2.6. Menu

MySmile provides two types of menu:

- Main menu — like “home”, “about”, “contact”, etc.
- Additional dynamic menu — internal links inside page for instance “readme”, “instructions”, etc.

Name	Required
Logo (Main Page) ¹	yes ²
Menu 1, ..., Menu 5 ³	no

¹Click on Logo is always follow to the first page from menu. Therefore changing menu order automatically change link for Logo.

²“Main page” is recommended page with slug name “index”.

³The optimal number of the main menu.

2.7. Content

All content can be changed via the [Admin panel](#).

2.7.1. Page parts

Part name	Description	l18n	Required	Unique
Menu	It is important to choose simple short name. It is used for menu.	yes	yes	no
Name	It is title of article. If it is not set “Menu” is used instead.	yes	no	no
Central Column	Central column of the article	yes	yes	no
Right Column: Text	Right Column of the article. Central column of the article	yes	no	no
Right Column: Multi-media	Photo or video that available in the Right Column	no	no	no
Bottom columns	Columns in the bottom side of the article. There are only 3 or less bottom columns	yes	no	no

2.7.2. SEO configuration

Search engine optimization (SEO) is presented as configuration parameters. Table bellow describe properties of such options.

Property	Description	l18n	Required	Unique
Meta tags	Title, description, keywords, photo description	yes	yes	no
Slug	Part of URL. All spaces and other non alphanumeric characters are changed to underscore.	no	yes	yes
Right Column, Photo	Alt property of tag	yes	yes ⁴	no

See also:

[Google recommendation](#) for SEO.

2.7.3. Technical configuration

Property	Description	l18n	Required	Unique
Statuses	draft, published	no	yes	no
Page type	Menu page, inner page, api page, api&menu page	no	yes	no
Date of creation	Date of creation page	no	yes	no
Date of modification	Date of last modification page	no	yes	no

⁴If Photo available

2.7.4. Preferences

Preferences contains information of contact and general technical information. All of them can be changed from the *Admin panel*.

Name	Default value	Description
Skype	myskype	
Email ⁵	myemail@email.com	
Phone number	000 000 000 00 00	
Turn on/off REST Api	True	
Max number of inner pages ⁶	3	Number of pages that can be shown under main menu when user follow links inside page.
Google Analytic code	000000	
Image quality	70	Global setting quality uploaded pictures. The picture quality is directly proportional to the size of the file. A value of 100 corresponds to the maximum quality.
Image autoscale	True	Global setting for autoscale uploaded pictures if its <i>width>333px</i> . Pictures will be scaled to <i>width=333px</i> .
Theme	modern	Choose a theme for switch

See also:

Technical configuration.

2.8. Admin panel

All content can be changed via the Admin panel.

2.8.1. Overview

Multilingual admin panel is available <http://127.0.0.1:8000/admin/>

- Username: *test*
- Password: *test*

⁵E-mail is automatically converted to the image for protection against spam bots.

⁶Number of pages of type **inner page** may be infinitely.

MySmile administration

Username:

Password:

Log in

English

Українська

English

MySmile v0.7.1 administration

Site administration

Authentication and Authorization	
Groups	+Add Change
Users	+Add Change
Pages	
Pages	+Add Change
Preferences	
Preferences	Change
Update	
Check update	

English






MySmile v0.7.0 administration Welcome, **test**. [View site](#) / [Change password](#) / [Log out](#)

[Home](#) > [Pages](#) > [Pages](#)

Select page to change

[Add page](#) +

Action: 0 of 11 selected

<input type="checkbox"/>	Slug	Status	3 ▲	Page type	1 ▼	Sort order	2 ▲	Photo thumb	Waiting for translation	Date update
<input type="checkbox"/>	index	published		api & menu page		10		 69.95K, WxH: 333x291px		27 April 2015, 18:00
<input type="checkbox"/>	two	published		menu page		20				10 April 2015, 14:09
<input type="checkbox"/>	three	published		menu page		30				13 April 2015, 19:15
<input type="checkbox"/>	four	published		menu page		40		 64.59K, WxH: 333x222px		10 April 2015, 14:10
<input type="checkbox"/>	five	published		menu page		50				10 April 2015, 14:10
<input type="checkbox"/>	item-one	published		inner page		100				13 April 2015, 18:58
<input type="checkbox"/>	item-two	published		inner page		110				06 July 2014, 15:19
<input type="checkbox"/>	item-three	published		inner page		120				10 April 2015, 14:10
<input type="checkbox"/>	item-four	published		inner page		130				10 April 2015, 14:10
<input type="checkbox"/>	item-five	published		inner page		140		 73.64K, WxH: 604x453px		10 April 2015, 14:11
<input type="checkbox"/>	item-six	published		inner page		150				04 July 2015, 14:33

[Home](#) > [Preferences](#) > [Preferences](#)

Select Preference to change

Name	Value
Email	myemail@email.com
Google Analytic code	0000000
Image autoscale	True
Image quality	70
Max number of inner pages	3
Phone number	000 000 000 00 00
Skype	myskype
Theme switcher	modern
Turn on/off REST Api	True

2.8.2. Language localisation

Add another language with option **LANGUAGES** in *mysmile/settings/base.py*. For example, this language will be Polish. Create directory *pl* into directory *mysmile/locale/*

Run command:

```
$ make makemessages lang=pl
```

Open and translate files *locale/pl/LC_MESSAGES/django.po*, then run::

```
$ make compilemessages lang=pl
```

2.9. Useful links

- http://ec.europa.eu/ipg/basics/legal/cookies/index_en.htm/

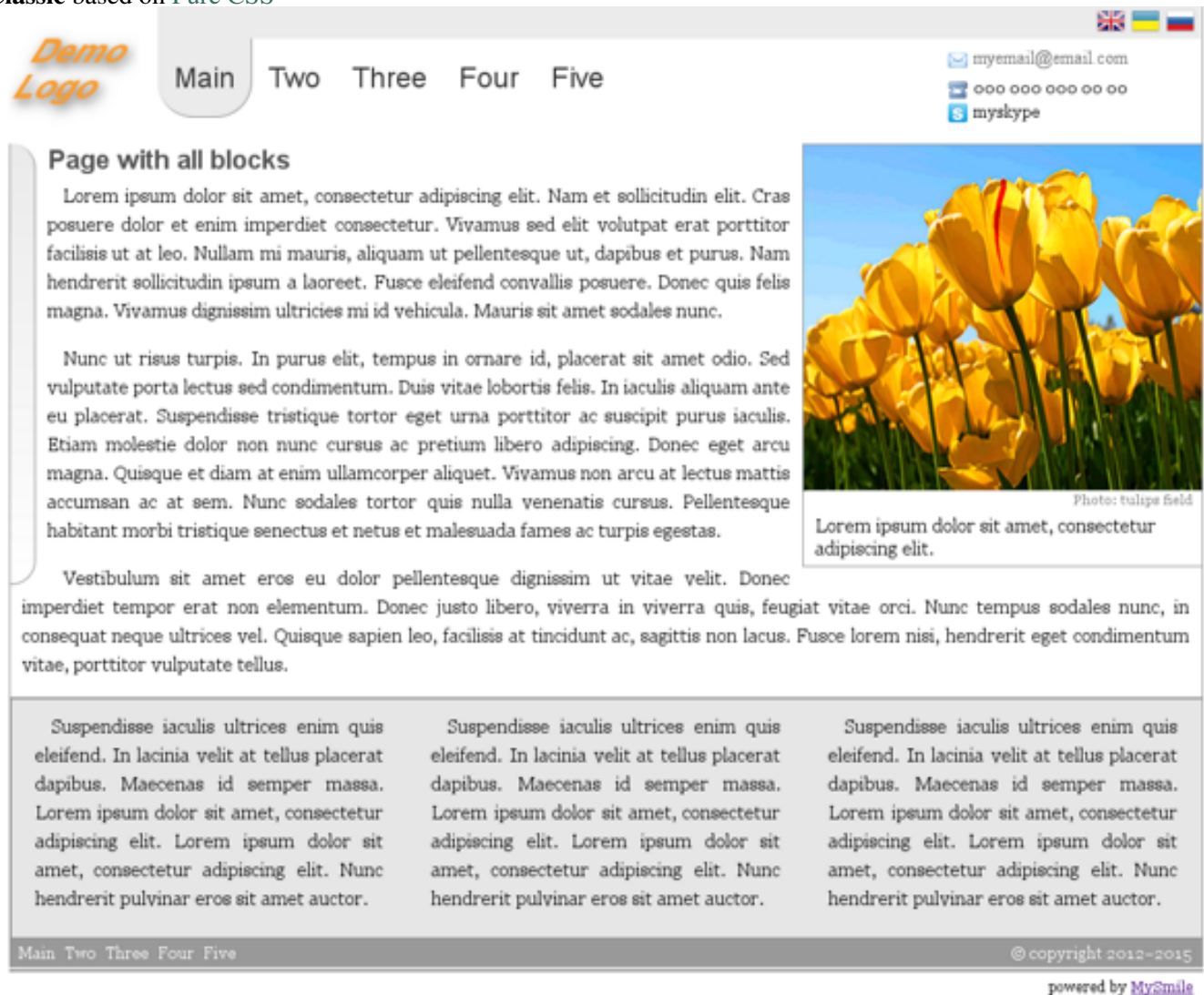
-
- http://en.wikipedia.org/wiki/Directive_on_Privacy_and_Electronic_Communications/
 - <http://www.python.org/>
 - <https://www.djangoproject.com/>
 - <https://docs.djangoproject.com/en/dev/intro/tutorial01/>

3. Themes

3.1. Available themes

Two fully responsive themes out of the box.

Classic based on [Pure CSS](#)



Modern based on [Bootstrap](#)



Page with all blocks

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam et sollicitudin elit. Cras posuere dolor et enim imperdiet consectetur. Vivamus sed elit volutpat erat porttitor facilisis ut at leo. Nullam mi mauris, aliquam ut pellentesque ut, dapibus et purus. Nam hendrerit sollicitudin ipsum a laoreet. Fusce eleifend convallis posuere. Donec quis felis magna. Vivamus dignissim ultricies mi id vehicula. Mauris sit amet sodales nunc.

Nunc ut risus turpis. In purus elit, tempus in ornare id, placerat sit amet odio. Sed vulputate porta lectus sed condimentum. Duis vitae lobortis felis. In iaculis aliquam ante eu placerat. Suspendisse tristique tortor eget urna porttitor ac suscipit purus iaculis. Etiam molestie dolor non nunc cursus ac pretium libero adipiscing. Donec eget arcu magna. Quisque et diam at enim ullamcorper aliquet. Vivamus non arcu at lectus mattis accumsan ac at sem. Nunc sodales tortor quis nulla venenatis cursus. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Vestibulum sit amet eros eu dolor pellentesque dignissim ut vitae velit. Donec imperdiet tempor erat non elementum. Donec justo libero, viverra in viverra quis, feugiat vitae orci. Nunc tempus sodales nunc, in consequat neque ultrices vel. Quisque sapien leo, facilisis at tincidunt ac, sagittis non lacus. Fusce lorem nisi, hendrerit eget condimentum vitae, porttitor vulputate tellus.



Photo: tulips field

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Suspendisse iaculis ultrices enim quis eleifend. In lacinia velit at tellus placerat dapibus. Maecenas id semper massa. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc hendrerit pulvinar eros sit amet auctor.

Suspendisse iaculis ultrices enim quis eleifend. In lacinia velit at tellus placerat dapibus. Maecenas id semper massa. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc hendrerit pulvinar eros sit amet auctor.

Suspendisse iaculis ultrices enim quis eleifend. In lacinia velit at tellus placerat dapibus. Maecenas id semper massa. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc hendrerit pulvinar eros sit amet auctor.

✉ myemail@email.com

☎ 000 000 000 00 00

💬 myskype

© 2015 MySmile, all right reserved

See also:

[How to add a new theme](#)

3.2. Configurations of theme

Property	Description	l18n	Required	Unique
Color	Color of active menu and vertical line on left side. Each page can have its own color vertical line on left side.	no	yes	no

Note: Property *Color* use in *classic* theme only

3.3. How to add a new theme

The new theme can be added manually by placing files in the appropriate directories

- apps/pages/templates/themes
- mysmile/static/themes

In this case the name of new theme is added automatically into admin panel. Switching between themes can be done using the *preferences*.

3.4. How to create a new theme

The new theme is created from any available template, which can be downloaded from the Internet.

4. REST API

4.1. REST API v1

This is version API based of Django.

4.1.1. Objective

REST API provides access to MySmile data. As a result it makes possible to create alternative UI design based for instance on [AngularJS](#) or even use as a part of [Wordpress](#). For such purpose PHP and Java clients were created (Python client is on a way).

4.1.2. Configuration

REST API is configurable feature in the MySmile project. Therefore to turn on/off it is need to change *Preferences* configuration parameter.

4.1.3. Endpoint

Endpoint of api is `http://mysmile.com.ua/api/`.

4.1.4. Methods

MySmile api is support only /GET method.

4.1.5. General parameters

List bellow show required parameters that have default values. It means that if they are not present in request that default value are going to be used.

Format

It sets format of request/response. Currently it is supported [JSON](#) but for future other types will be acceptable [XML](#) or [JSONP](#).

parameter name	value	default
format	json	json

Version monitoring

Parameter that shows what version of api is used.

parameter name	value	default
v	1	1

Note: This parameter is not used now. It is reserved for in future need for version migration politics.

Language

Parameter that sets language of response.

Parameter name	value	default
lang	en	en ¹

4.1.6. Request

Request format is: endpoint + resource + required parameters. General parameters can be omitted because they have default values. For instance request to get content with slug *home* in *json* format with *en* language:

```
/content?slug=home&v=1&lang=en&format=json
```

The same but with omitted general parameters:

```
/content?slug=home
```

Request to get list of all contents:

```
/content
```

See *Examples* for details.

4.1.7. Response

Response contains code and body (accordingly HTTP specification). Body has duplicated HTTP code value.

JSON

Success

General format of success response is:

```
{ code: 200, data: {array or object} }
```

Example:

```
{ code: 200, data: ["home", "about", "price", "contact"] }
```

Failure

General format of failure response is:

```
{ code: HTTP code, msg: "Error message" }
```

Example:

```
{ code: 500, msg : "Internal server error" }
```

4.1.8. Resources

Those parameters should be presented in the every request otherwise default values will be used.

¹The default value is the first order LANGUAGES from *settings/base.py*

Content

That resource returns content of current page by slug or list of available contents if slug is not set.

Resource	GET ²	Example
/content	list of contents	<pre>{ code: 200, data: [{"home": "Home"}, {"about": "About"}, {"contact": "Contact"}] }</pre>
/content?slug=foo	current content by slug	<pre>{ code: 200, data: { menu: "Home", name: "name of central column", col_central: "html of central column", col_right: "html of right column", col_bottom: ["html of first bottom column", "html of second bottom column", "html of third bottom column"], youtube: "link to YouTube resource", photo: { src: "url to photo", alt: "alt text", description: "description" } } }</pre>

Language

That resource returns list of available languages.

Resource	GET	Example
/language	list of supported languages	<pre>{ code: 200, data: ["ua", "ru", "en"] }</pre>

Contact

That resource ³ returns structured contact information such as phone, email, skype.

³ Parameters of response configurable through MySmile settings table in database (since v 0.5.0 of api sever).

Resource	GET	Example
/contact	list of contact data	<pre>{ code: 200, data: { email: "test@mysmile.com", phone: "111-11-11", skype: "mysmile" } }</pre>

Notes

4.1.9. Errors

HTTP code	Description	Example ⁴
403	Request was a valid one, but server was refusing to respond to it.	<pre>{ code: 403, msg: "Forbidden" }</pre>
404	Resource not found.	<pre>{ code: 404, msg: "Not Found" }</pre>
405	A request was made of a resource using a request method not supported by that resource. Try to get access to POST if only GET available.	<pre>{ code: 405, msg: "Method Not Allowed" }</pre>
500	A generic error message, given when an unexpected condition was encountered and no more specific message is suitable. Error in the code or invalid parameters use.	<pre>{ code: 500, msg: "Internal Server Error" }</pre>
501	The server either does not recognize the request method, or it lacks the ability to fulfill the request. Resource does not implemented	<pre>{ code: 501, msg: "Not Implemented" }</pre>

4.1.10. Examples

Request/GET	http://mysmile.com.ua/api/content?v=1&lang=en&format=json
Response/Code	200
Response/Header	Content-Type application/json;
Response/Body	<pre>{ code: 200, data: [{"home": "Home"}, {"about": "About"}, {"contact": "Contact"}] }</pre>

4.1.11. Useful links

1. Apigee: <http://www.youtube.com/watch?v=QpAhXa12xvU>
2. Http code:
 - http://en.wikipedia.org/wiki/List_of_HTTP_status_codes
 - http://www.w3schools.com/tags/ref_httpmessages.asp
3. JSON: <http://www.json.org/>

5. Frequently Asked Questions

Can I use the MySQL database instead of SQLite?

Yes. To do this, do the following:

1. Run `$ pip3 install pymysql==0.6.7`
2. Add into *manage.py* this lines before line `if __name__ == "__main__":`:

```
import pymysql
pymysql.install_as_MySQLdb()
```

3. Set Django DATABASES settings in *config/local.py* like

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'mysmile',
        'USER': 'root',
        'PASSWORD': 'password',
        'HOST': '127.0.0.1',
        'PORT': '3306',
    }
}
```

4. Create database *mysmile*
5. Run command `$ make newdb`
6. Run local server `$ make run`

6. Glossary

additional menu Additional menu is a dynamic navigation that is generated from inner pages after last was visited. Configuration key “MAX_INNERLINK_HISTORY” set a max number of such pages in menu.

See also:

Preferences.

api page Api page is specify pages that are available only from REST API.

See also:

Technical configuration.

api & menu page Api & menu page is an combination of **api page** and **menu page**.

See also:

Technical configuration.

inner page Inner page is characterised pages that used as link inside of **menu page**. Such kind of pages are not part of main menu but from visited inner pages are formed the **additional menu**.

See also:

Technical configuration.

menu page Menu page is described pages that are shown in the main menu.

See also:

Technical configuration.

Index

A

additional menu, [25](#)
api & menu page, [25](#)
api page, [25](#)

I

inner page, [25](#)

M

menu page, [25](#)