python-textunited

Release 0.1.0

Nov 27, 2017

Contents

CHAPTER 1

Overview

docs	
tests	
package	

A wrapper around the Text United API written in Python. Each Text United object is represented by a corresponding Python object. The attributes of these objects are cached, but the child objects are not.

- Free software: GNU General Public License v3 or later (GPLv3+)

1.1 Installation

```
pip install python-textunited
```

1.2 Documentation

https://python-textunited.readthedocs.io/

1.3 Development

To run the all tests run:

```
tox
```

Note, to combine the coverage data from all the tox environments run:

Windows	`set PYTEST_ADDOPTS=--cov-append` `tox`
Other	`PYTEST_ADDOPTS=--cov-append tox`

CHAPTER 2

Installation

At the command line:

```
pip install python-textunited
```

CHAPTER 3

Usage

To use python-textunited in a project:

```
import textunited
```

3.1 Authenticate

This example show how to get the client. This client will be used on the examples below

```
from textunited import TextUnitedClient

client = TextUnitedClient(company_id='123', api_key='abc')
```

3.2 List all account

```
account_list = client.list_accounts()

for account in account_list:
    print(account)

# it will show something similar:
# id#001 user001@example.com
# id#002 user002@example.com
# id#003 user003@example.com
```

3.3 List all projects and get files content

In this example, it is showed how to list projects and access to the to the file content. For more info, please check the documentation inside of each model.

```
projects_list = client.list_projects()

for project in projects_list:
    print(project)

# it will show something similar to:
# id#0001 "example1" EN -> AR by Jonh Example (Completed)
# id#0002 "example2" EN -> AR by Jonh Example (In preparation)


# Get files inside project. Content is not downloaded, only the metadata
file_list = projects_list[0].get_files()
file = file_list[0]

# if the status is Translated, we can get translated files
file.get_translated_content()
file.translated_content

# Also, it is possible to download the source content
file.get_source_content()
file.source_content
```

3.4 Create a new project

A project is composed by multiple files to be translated, each file needs to be encapsulated in a FileUpload class and pass in a list to Project Request.

```
from textunited import FileUpload, ProjectRequest

file1 = FileUpload(name='example1.txt', b'hello world')
file2 = FileUpload(name='example1.txt', b'hello world')

project_request = ProjectRequest(
    name='Project 1',
    source_language_id=12,
    target_language_id=13,
    description='This is a description',
    files=[file1, file2],
    translator_id=12,
)

project_id = client.add_project(project_request)

client.get_project(project_id)
```

CHAPTER 4

Supported languages

In this moment, the language supported by this library are the following ones:

- ar_ae = 21 # Arabic (United Arab Emirates)
- de_de = 53 # German (Germany)
- en_ca = 158 # English (Canada)
- en_gb = 40 # English (UK)
- en_us = 41 # English (US)
- es_co = 108 # Spanish (Colombia)
- es_es = 104 # Spanish (Spain)
- fr_ca = 48 # French (Canada)
- ja = 72 # Japanese
- ko = 76 # Korean
- pt_br = 94 # Portuguese (Brazil)
- zh_hant = 33 # Chinese (Traditional)
- zn_hans = 32 # Chinese (Simplified)

Text United supports the following languages. Those languages can be enabled by adding the language code and the identifier in languages.py.

- Abkhaz
- Afrikaans
- Albanian
- Amharic
- Arabic (Algeria)
- Arabic (Bahrain)

- Arabic (Egypt)
- Arabic (General)
- Arabic (Iraq)
- Arabic (Jordan)
- Arabic (Kuwait)
- Arabic (Lebanon)
- Arabic (Libyan Arab Jamahiriya)
- Arabic (Morocco)
- Arabic (Oman)
- Arabic (Qatar)
- Arabic (Saudi Arabia)
- Arabic (Sudan)
- Arabic (Syrian Arab Republic)
- Arabic (Tunisia)
- Arabic (United Arab Emirates)
- Arabic (Yemen)
- Armenian
- Assamese
- Azerbaijani
- Basque
- Belarusian
- Bengali
- Bodo
- Bosnian
- Bulgarian
- Catalan
- Chinese
- Chinese (Simplified)
- Chinese (Traditional)
- Croatian
- Czech
- Danish
- Dogri
- Dutch (Belgium)
- Dutch (Netherlands)
- English (UK)

- English (US)
- Esperanto
- Estonian
- Faroese
- Fijian
- Finnish
- French (Belgium)
- French (Canada)
- French (France)
- French (Swiss)
- Galician
- Georgian
- German (Austria)
- German (Germany)
- German (Lichtenstein)
- German (Luxembourg)
- German (Swiss)
- Greek
- Greenlandic
- Guarani
- Gujarati
- Haitian Creole
- Hausa
- Hebrew
- Hindi
- Hungarian
- Icelandic
- Igbo
- Indonesian
- Irish (Gaelic)
- Italian
- Italian (Swiss)
- Japanese
- Javanese
- Kannada
- Kashmiri

- Kazakh
- Kirghiz (Kyrgyz)
- Konkani
- Korean
- Kurdish
- Lao
- Latin
- Latvian
- Lithuanian
- Macedonian
- Maithili
- Malay/Malaysian
- Malayalam
- Maltese
- Marathi
- Meitei (Manipuri)
- Moldovan
- Mongolian
- Montenegrin
- Nepali
- Norwegian
- Odia
- Pashto
- Persian
- Polish
- Portuguese (Brazil)
- Portuguese (Portugal)
- Punjabi
- Romanian
- Russian
- Sanskrit
- Santali
- Scottish Gaelic
- Serbian
- Sindhi
- Slovak

- Slovenian
- Somali
- Spanish (Argentina)
- Spanish (Bolivia)
- Spanish (Chile)
- Spanish (Colombia)
- Spanish (Costa Rica)
- Spanish (Dominican Republic)
- Spanish (El Salvador)
- Spanish (Equador)
- Spanish (Guatemala)
- Spanish (Honduras)
- Spanish (Mexico)
- Spanish (Nicaragua)
- Spanish (Panama)
- Spanish (Paraguay)
- Spanish (Peru)
- Spanish (Puerto Rico)
- Spanish (Spain)
- Spanish (US)
- Spanish (Uruguay)
- Spanish (Venezuela)
- Swahili
- Swedish
- Tagalog
- Tamil
- Tatar
- Telugu
- Thai
- Tibetan
- Turkish
- Turkmen
- Ukrainian
- Urdu
- Uzbek
- Vietnamese

- Welsh
- Xhosa
- Yoruba
- Zulu

CHAPTER 5

Reference

CHAPTER 6

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

6.1 Bug reports

When reporting a bug please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

6.2 Documentation improvements

python-textunited could always use more documentation, whether as part of the official python-textunited docs, in docstrings, or even on the web in blog posts, articles, and such.

6.3 Feature requests and feedback

The best way to send feedback is to file an issue at https://github.com/tenplatform/python-textunited/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that code contributions are welcome :)

6.4 Development

To set up *python-textunited* for local development:

1. Fork python-textunited (look for the "Fork" button).
2. Clone your fork locally:

```
git clone git@github.com:your_name_here/python-textunited.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

4. When you're done making changes, run all the checks, doc builder and spell checker with tox one command:

```
tox
```

5. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

6.4.1 Pull Request Guidelines

If you need some code review or feedback while you're developing the code just make the pull request.

For merging, you should:

1. Include passing tests (run `tox`)[1].
2. Update documentation when there's new API, functionality etc.
3. Add a note to `CHANGELOG.rst` about the changes.
4. Add yourself to `AUTHORS.rst`.

6.4.2 Tips

To run a subset of tests:

```
tox -e envname -- py.test -k test_myfeature
```

To run all the test environments in *parallel* (you need to `pip install detox`):

```
detox
```

[1] If you don't have all the necessary python versions available locally you can rely on Travis - it will run the tests for each change you add in the pull request.
It will be slower though ...

CHAPTER 7

Authors

- Ten Product - https://www.tengroup.com

CHAPTER 8

Changelog

8.1 0.1.0 (2017-10-17)

- First release with the main modules and test working.

CHAPTER 9

Indices and tables

- genindex
- modindex
- search