faker-file-qt Documentation

Release 0.1

Artur Barseghyan <artur.barseghyan@gmail.com>

Sep 26, 2023

CONTENTS

1 Prerequisites 3

2 Documentation 5

3 Installation 7
- 3.1 Latest stable version from PyPI 7
- 3.2 Or development version from GitHub 7
- 3.3 Install locally in development mode 7

4 Running 9

5 Testing 11

6 Related projects 13

7 Writing documentation 15

8 License 17

9 Support 19

10 Author 21

11 Project documentation 23
- 11.1 Security Policy 23
  - 11.1.1 Reporting a Vulnerability 23
  - 11.1.2 Supported Versions 24
- 11.2 Contributor guidelines 24
  - 11.2.1 Developer prerequisites 24
    - 11.2.1.1 pre-commit 24
  - 11.2.2 Code standards 24
  - 11.2.3 Requirements 24
  - 11.2.4 Virtual environment 25
  - 11.2.5 Documentation 25
  - 11.2.6 Testing 25
  - 11.2.7 Pull requests 25
  - 11.2.8 Questions 26
  - 11.2.9 Issues 26
- 11.3 Release history and notes 26
  - 11.3.1 0.1.4 26
  - 11.3.2 0.1.3 26

11.3.3 0.1.2 26
11.3.4 0.1.1 27
11.3.5 0.1 27
11.4 Package 27
11.5 Indices and tables 27

PyQT UI for faker-file.

CHAPTER

ONE

PREREQUISITES

- faker-file (faker-file[common])
- QDarkStyle
- PyQT5

CHAPTER

TWO

DOCUMENTATION

- Documentation is available on Read the Docs.

CHAPTER

THREE

INSTALLATION

3.1 Latest stable version from PyPI

```
pipx install faker-file-qt
```

3.2 Or development version from GitHub

```
pipx install https://github.com/barseghyanartur/faker-file-qt/archive/main.tar.gz
```

3.3 Install locally in development mode

```
pip install -e .[dev]
```

CHAPTER

FOUR

RUNNING

```
faker-file-qt
```

CHAPTER

FIVE

TESTING

Simply type:

```
pytest -vrx
```

Or use tox:

```
tox
```

CHAPTER

SIX

RELATED PROJECTS

Check the demo(s):

- REST API demo (based on faker-file-api REST API)
- UI frontend demo (based on faker-file-ui UI frontend)
- WASM frontend demo (based on faker-file-wasm WASM frontend)

CHAPTER

SEVEN

WRITING DOCUMENTATION

Keep the following hierarchy.

```
=====
title
=====

header
======

sub-header
----------

sub-sub-header
~~~~~~~~~~~~~~

sub-sub-sub-header
^^^^^^^^^^^^^^^^^^

sub-sub-sub-sub-header
++++++++++++++++++++++

sub-sub-sub-sub-sub-header
**************************
```

CHAPTER EIGHT

LICENSE

MIT

CHAPTER NINE

SUPPORT

For security issues contact me at the e-mail given in the *Author* section.

For overall issues, go to GitHub.

CHAPTER TEN

AUTHOR

Artur Barseghyan <artur.barseghyan@gmail.com>

CHAPTER

ELEVEN

PROJECT DOCUMENTATION

Contents:

Table of Contents

- *faker-file-qt*
  - *Prerequisites*
  - *Documentation*
  - *Installation*
    - *Latest stable version from PyPI*
    - *Or development version from GitHub*
    - *Install locally in development mode*
  - *Running*
  - *Testing*
  - *Related projects*
  - *Writing documentation*
  - *License*
  - *Support*
  - *Author*
  - *Project documentation*

11.1 Security Policy

11.1.1 Reporting a Vulnerability

Do not report security issues on GitHub!

Please report security issues by emailing Artur Barseghyan <artur.barseghyan@gmail.com>.

11.1.2 Supported Versions

Make sure to use the latest version.

The two most recent `faker-file-qt` release series receive security support.

For example, during the development cycle leading to the release of `faker-file-qt` 0.17.x, support will be provided for `faker-file-qt` 0.16.x.

Upon the release of `faker-file-qt` 0.18.x, security support for `faker-file-qt` 0.16.x will end.

Version	Supported
0.1.x	Yes

11.2 Contributor guidelines

11.2.1 Developer prerequisites

11.2.1.1 pre-commit

Refer to pre-commit for installation instructions.

TL;DR:

```
pip install pipx --user  # Install pipx
pipx install pre-commit  # Install pre-commit
pre-commit install  # Install pre-commit hooks
```

Installing pre-commit will ensure you adhere to the project code quality standards.

11.2.2 Code standards

black, isort, ruff and doc8 will be automatically triggered by pre-commit. Still, if you want to run checks manually:

```
make black
make doc8
make isort
make ruff
```

11.2.3 Requirements

Requirements are installed as follows.

```
make install
```

11.2.4 Virtual environment

You are advised to work in virtual environment.

TL;DR:

```
python -m venv env
pip install -e .
pip install -r examples/requirements/django_3_2_and_flask.txt
```

11.2.5 Documentation

Check documentation.

11.2.6 Testing

Check testing.

If you introduce changes or fixes, make sure to test them locally using all supported environments. For that use tox.

```
make test
```

In any case, GitHub Actions will catch potential errors, but using tox speeds things up.

11.2.7 Pull requests

You can contribute to the project by making a pull request.

For example:

- To fix documentation typos.
- To improve documentation (for instance, to add new recipe or fix an existing recipe that doesn't seem to work).
- To introduce a new feature (for instance, add support for a non-supported file type).

Good to know:

- Test suite makes extensive use of parametrization. Make sure you have added your changes in the right place.

General list to go through:

- Does your change require documentation update?
- Does your change require update to tests?
- Did you test both Latin and Unicode characters?
- Does your change rely on third-party cloud based service? If so, please make sure it's added to tests that should be retried a couple of times. Example: `@pytest.mark.flaky(reruns=5)`.

When fixing bugs (in addition to the general list):

- Make sure to add regression tests.

When adding a new feature (in addition to the general list):

- Check the licenses of added dependencies carefully and make sure to list them in prerequisites.
- Make sure to update the documentation.

11.2.8 Questions

Questions can be asked on GitHub discussions.

11.2.9 Issues

For reporting a bug or filing a feature request use GitHub issues.

Do not report security issues on GitHub. Check the support section.

11.3 Release history and notes

Sequence based identifiers are used for versioning (schema follows below):

```
major.minor[.revision]
```

- It's always safe to upgrade within the same minor version (for example, from 0.3 to 0.3.4).
- Minor version changes might be backwards incompatible. Read the release notes carefully before upgrading (for example, when upgrading from 0.3.4 to 0.4).
- All backwards incompatible changes are mentioned in this document.

11.3.1 0.1.4

2023-09-23

- Set minimal required version of faker-file to 0.17.8.
- Add selects for image, pdf and mp3 generator selection.

11.3.2 0.1.3

2023-07-28

- Set minimal required version of faker-file to 0.17.2.
- Add menus.

11.3.3 0.1.2

2023-07-26

- UI improvements.
- Added JSON file provider.
- Improved tests.

11.3.4 0.1.1

2023-07-25

- UI improvements.
- Open default app on click on the results.

11.3.5 0.1

2023-07-24

- Initial beta release.

11.4 Package

11.5 Indices and tables

- genindex
- modindex
- search